NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

AN ANALYSIS OF THREE AVCAL INVENTORY MODELS
USING THE TIGER SIMULATION MODEL

by

Mark David Sullivan

September 1984

Thesis Advisor: F.R. Richards

Approved for public release; distribution unlimited.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Analysis of Three AVCAL Inventory Models Using the TIGER Simulation Model		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; September 1984 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Mark David Sullivan		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California		12. REPORT DATE September 1984 13. NUMBER OF PAGES 157
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

AVCAL
RIMAIR
ACIM
simulation
inventory models
TIGER
operational availability

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This thesis investigates the effectiveness of three AViation Consolidated Allowance List (AVCAL) inventory models in achieving aircraft system operational availability. The three models studied are the Aviation Supply Office (ASO) Model, the Repairables Integrated Model for Aviation (RIM-AIR), and the Availability Centered Inventory Model (ACIM). TIGER, a simulation model developed

by Naval Seas Systems Command, is amended to accommodate simulation of multiple aircraft sorties with a realistic parts pipeline operation. AVCAL model inventory levels are compared over a ninety day period utilizing availability statistics computed by TIGER.

Approved for public release; distribution unlimited

An Analysis of Three AVCAL Inventory Models Using the TIGER Simulation Model

by

Mark David Sullivan
Lieutenant Commander, United States Navy
B.S., University of Louisville, 1974

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL

September 1984

ABSTRACT

This thesis investigates the effectiveness of three AViation Consolidated Allowance List (AVCAL) inventory models in achieving aircraft system operational availability. The three models studied are the Aviation Supply Office (ASO) Model, the Repairables Integrated Model for Aviation (RIM-AIR), and the Availability Centered Inventory Model (ACIM). TIGER, a simulation model developed by Naval Seas Systems Command, is amended to accommodate simulation of multiple aircraft sorties with a realistic parts pipeline operation. AVCAL model inventory levels are compared over a ninety day period utilizing availability statistics computed by TIGER.

TABLE OF CONTENTS

I. INTRODUCTION ---------------------------------- 11

A. BACKGROUND ---------------------------------- 11

B. PURPOSE ---------------------------------- 12

C. AIRCRAFT DATA ---------------------------------- 12

D. MAJOR TOPICS ---------------------------------- 13

II. THE TIGER SIMULATION PROGRAM ------------------- 15

A. INTRODUCTION ---------------------------------- 15

B. MAIN FEATURES OF TIGER --------------------- 15

1. Simulation ---------------------------------- 15

2. TIGER Statistics ------------------------- 17

3. TIGER Subroutines ------------------------ 22

C. TIGER CHANGES ---------------------------------- 23

1. Aircraft Sortie Simulation ------------- 23

2. Equipment: Repair and Resupply -------- 24

3. New TIGER Subroutines ------------------ 25

4. TIGER Validation ------------------------ 26

III. ASO MODEL-- 31

A. MODEL DESCRIPTION -------------------------- 31

B. INVENTORY DETERMINATION -------------------- 33

1. Attrition Rules -------------------------- 35

2. Rotatable Pool Rules -------------------- 36

C. MODEL LIMITATIONS -------------------------- 37

IV. RIMAIR MODEL ---------------------------------- 40

A. MODEL DESCRIPTION --------------------------- 40

B. INVENTORY DETERMINATION -------------------- 42

1. Steady-state Supply Effectiveness------- 42

2. Optimization --------------------------- 45

3. External Constraints ------------------- 50

C. MODEL LIMITATIONS --------------------------- 51

V. ACIM MODEL ------------------------------------ 54

A. MODEL DESCRIPTION --------------------------- 54

B. INVENTORY DETERMINATION -------------------- 56

1. ACIM Solution Equations ---------------- 56

2. Objective Function --------------------- 58

3. Input Data ----------------------------- 62

C. MODEL LIMITATIONS --------------------------- 71

VI. TEST RESULTS ---------------------------------- 74

A. INTRODUCTION -------------------------------- 74

B. FIXED BUDGET ANALYSIS ---------------------- 77

C. VARIABLE BUDGET ANALYSIS ------------------- 81

D. VARIABLE MSRT ANALYSIS --------------------- 83

VII. SUMMARY, CONCLUSIONS AND RECOMMENDATIONS ------- 87

A. SUMMARY AND CONCLUSIONS ------------------- 87

B. RECOMMENDATIONS ---------------------------- 90

APPENDIX A: TIGER DATA CARD FORMATS ------------------ 92

APPENDIX B: TIGER INPUT DATA FOR SERIES SYSTEM ------- 103

APPENDIX C: TIGER INPUT DATA FOR PARALLEL SYSTEM------ 106

APPENDIX D: TIGER PROGRAM LISTING -------------------- 109

LIST OF REFERENCES -- 155

INITIAL DISTRIBUTION LIST -------------------------------- 157

LIST OF TABLES

I. Critical Equipment Computation -------------------- 20

II. TIGER Validation Results ---------------------------- 30

III. Part Parameters ------------------------------------ 75

IV. Fixed Budget Summary -------------------------------- 79

V. Critical Equipment Analysis of ASO Model ---------- 80

VI. AVCAL Stock Levels for Variable Budget ------------ 83

VII. RIMAIR vs. ACIM Performance for Variable Budget---- 84

VIII. ACIM Performance for Variable MSRT ---------------- 86

LIST OF FIGURES

2.1 Phase Sequence Configuration ------------------- 23

2.2 Part Pipelines ---------------------------------- 24

3.1 ASO Repair/Resupply Model ---------------------- 32

3.2 TAT Elements ------------------------------------ 38

4.1 RIMAIR Stockage Level Options ------------------ 49

5.1 Format A Data Elements and Record Example ------ 63

5.2 Format L Data Elements and Record Example ------ 67

5.3 Format I Data Elements and Record Example ------ 69

6.1 System Configuration (Mission Code D) ---------- 76

ACKNOWLEDGEMENTS

I wish to thank Peter Evanovich and Barbara Measell, from the Center for Naval Analyses, for their patient help in data collection and model research. I also wish to thank my thesis advisor, Professor Russell Richards, for his helpful guidance through the entire evolution of this thesis. A special thanks goes to my wife, Mary, and my children, Christine, Timothy, and Sean. Only with their constant support and understanding was I able to complete this work.

I. INTRODUCTION

A. BACKGROUND

Navy ships, Marine Air Groups (MAG) and shore activities receive AViation Coordinated Allowance Lists (AVCAL) to support assigned aircraft for a prescribed period of time (usually ninety days for ship and MAGs). AVCALS for carrier deployed squadrons are especially crucial because operational commitments will often exceed projected flight time estimates and because the repair/resupply pipeline can be extremely lengthy. These lists are produced by Aviation Supply Office (ASO) prior to the assignment of aviation elements to a ship or MAG. AVCAL's consist of Allowance Requirements Registers (ARR's), which contain the projection of the range (which parts?) and depth (how many?) of spare assemblies and parts necessary to support the aircraft and associated support equipment at the Organizational (O) and Intermediate (1) levels.

One of the key measures of effectiveness of a squadron's performance is the aircraft operational availability. Operational availability has many definitions, but here availability refers to the expected percentage of time that a weapon system or individual equipment will be ready to perform satisfactorily in an operating environment.

The Organizational (squadron) level maintenance concept is based on "remove and replace". Weapon Replaceable Assemblies (WRAs) are designed for rapid removal from the aircraft. Parts that can be repaired at the intermediate level are inducted to the Aviation Intermediate Maintenance Department (AIMD) on the carrier. If a spare part is available, a replacement part is issued to the squadron. Parts that cannot be repaired on-ship, are sent off-ship to the depot level repair facility. A replacement part is re-ordered for the part placed in the off-ship supply pipeline.

B. PURPOSE

This study will examine three inventory models currently used to determine AVCALs. Model effectiveness will be compared using simulated aircraft systems, representing systems found on Navy E-2C Hawkeye aircraft. These parts are also items that are found on the E-2C Mission-Essential Subsystem Matrices (MESM), OPNAV Instruction 5442.4H [Ref. 1]. After inventory levels are computed, operational availability is estimated by simulation of aircraft flights on the TIGER program.

C. AIRCRAFT DATA

This thesis will concentrate only on repairable parts, although consumables are also normally included in AVCAL computation. Parts included in this study are WRAs from

the avionics portion of the E-2C aircraft, coded for removal at the squadron level and repairable at the intermediate level. The majority of the parts are complicated, expensive pieces that cannot be stocked indiscriminately at high levels.

Equipment data was taken from Center for Naval Analyses computer tapes of Navy wide E-2C parts data for the year 1981. Item unit costs, failure rates, and BCM (beyond the capability of maintenance) rates reflect 1981 levels. Only a limited number of parts were considered due to the limitations of the TIGER simulation program.

AVCAL budget levels are based on predicted quarterly aircraft operating hours. However, a recent E-2C squadron operating level exceeded 1500 total hours in a quarter of high tempo operations on deployment, thus exceeding historical operating levels by almost 50%. This is not uncommon and suggests that it is important to have inventory levels designed to achieve maximum availability while meeting imposed budget constraints.

D. MAJOR TOPICS

Chapter II discusses the TIGER simulation model used to compare inventory level effectiveness. The TIGER model is examined, along with major changes introduced to the model for this study. TIGER is a flexible program that allows for sensitivity analysis by easy modification of part parameters and system configuration. Aircraft sorties are simulated

over a period of ninety days and the resulting system availability is calculated.

Chapter III outlines one of the major inventory models presently used to compute AVCAL, the ASO Manual Model. Chapter IV continues with an outline of the RIMAIR Model, and Chapter V covers the ACIM Model. Chapter VI presents test results for the three models studied and Chapter VII presents a thesis summary, conclusions derived from the analysis, and recommendations.

II. THE TIGER SIMULATION PROGRAM

A. INTRODUCTION

TIGER is the generic name for a family of computer programs developed for Naval Sea Systems Command in 1979 which can be used to evaluate, by simulation, a complex system in order to estimate various reliability, readiness, and availability measures. Originally designed for testing ship and shipboard weapon systems, TIGER has been amended several times at the Naval Postgraduate School. Major changes were undertaken by J. Leather in 1980 [Ref. 2], and P. O'Reilly in 1981 [Ref. 3].

During the course of this study several significant changes were made to the TIGER program. Several subroutines were changed and two subroutines were added. This chapter will outline the general features of TIGER and then detail the changes made in this study. The TIGER Manual [Ref. 4] is the primary reference source for all input, output, and optional features contained in the TIGER program. Only those options pertinent to this study will be outlined here.

B. MAIN FEATURES OF TIGER

1. Simulation

TIGER uses Monte Carlo simulation techniques to evaluate the system model under consideration. Random

numbers drawn from Naval Postgraduate School's LLRANDOMII [Ref. 5] were used to generate equipment failure times, repair times and other random numbers used in the simulation. Based on the system configuration of equipment, the system up and down times were determined. Based on these times, system measures of performance were calculated. The simulation was repeated a specified number of times and the results averaged.

The configuration of the system being modeled is defined in a top-down breakdown of the system into subsystem(s), groups and equipments. Each type of equipment is given a unique identifying number and its characteristics (MTBF, MITTR, BCM rate, unit cost) are stated.

Events are significant mission occurrences. TIGER recognizes the following types of events:

Equipment Failure (UP to DOWN)
Equipment Repair (DOWN to UP)
End of Phase Period Within Mission
Beginning of Mission
End of Mission

These five types of events are stored in sequential order according to time occurrence. The first event becomes the next step at which computations within TIGER are done.

The concept of phases is essential to the operation of TIGER. A phase is s specified length of time that is characterized by a set of equipment operating rules. For this study two phase types were utilized:

Phase Type (1) is the flight phase. Equipments were subject to failure during this phase. Parts that failed during the flight phase could not be repaired until the beginning of the next ondeck phase. Parts being repaired ondeck continued to be repaired. If one part in the aircraft system failed during flight, other parts on the aircraft were still subject to failure.

Phase type (2) is the ondeck phase. Equipments were not subject to failure during the ondeck (repair) phase. Aircraft parts that failed during the previous flight phase were taken off the aircraft, replaced with a spare if available, and the failed part was placed in the repair pipeline. If no spare was available the aircraft system was considered to be in a degraded mode; that is, flight was possible but system capabilities were decreased depending on the essentiality of the failed parts.

Parts in the repair pipeline continue to be ordered, shipped, and repaired during all phases.

2. TIGER Statistics

The statistics calculated by TIGER are system reliability, readiness, and availability. The reliability estimator used in TIGER is the ratio of the number of successful missions to the total number of attempted missions. A successful mission occurs when no system failure occurs during the course of the mission. For a system composed of high failure rate parts such as those in

this study, there is no statistical chance of completing a mission of ninety days without a system failure. For this reason the reliability statistic was not used.

The average readiness estimator (RED (EST)) used in TIGER is the ratio of readiness (RED) uptime during the entire mission to total calendar time of the entire mission.

Red uptime = Calendar time - Red downtime

Red downtime = Downtime prior to mission abort + time after mission abort

This statistic was not used because it provides no indication of system availability after the first system failure.

The most informative statistic is the average availability, AVA AVERAGE (EST), or simply AVA. The availability parameter is the probability that the system will be in a satisfactory operating condition. It is estimated in TIGER as:

$$AVA = \frac{\text{Total Uptime for all phases}}{\text{Total Simulation Time}}$$

For the scenario used in this paper AVA would include downtime for an aircraft system during both the flight phase and the ondeck phase. Since this study is primarily concerned with aircraft system availability airborne, and it is assumed that the scheduled launches would continue even with degraded systems (a more likely event than waiting on deck for 100% system availability), a new statistic was introducted.

The new availability parameter, AVMUP, measures the availability of the aircraft system only during flight phases.

$$\text{AVMUP} = \frac{\text{Summation of all Flight Phase Uptime}}{\text{Summation of Total Flight Phase Time}}$$

Although AVMUP approximates the AVA value, the only time the two statistics are equal is when the ratio, A; where

$$\text{A} = \text{Flight Phase Uptime/Flight Phase Downtime}$$

is equal to the ratio, B; where

$$\text{B} = \frac{\text{Repair (ondeck) Phase Uptime}}{\text{Repair (ondeck) Phase Downtime}}$$

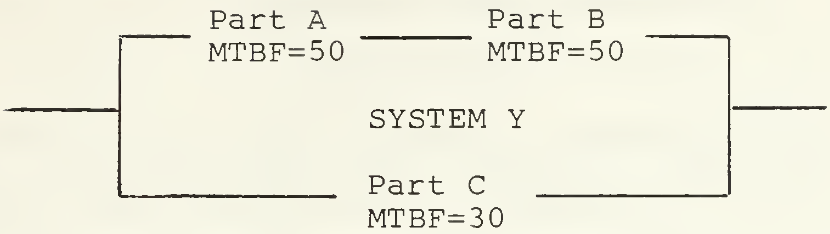
There are many scenarios in which these two ratios will not be equal. For example, a system with high failure rate parts will tend to have a lower A ratio. But these same parts may not decrease ratio B to the same degree if adequate spares are available. AVMUP emphasizes part criticality and reliability more than AVA. For this study, AVMUP was consistently several percentage points less than AVA.

Another set of statistics used in this thesis was the Critical Equipments Summary produced by TIGER. This is an optional printout that points out parts that are "worst offenders". Parts that caused the system to go to a down status or parts that failed while the system was already in a down status are listed in this output.

Table I provides an example of this summary. It depicts the events in a 90 hour mission, using a three part system, System Y. At time 8.96 hours, Part A fails, but since System Y is still up, no system downtime is recorded. At time 34.04 hours, part B fails, causing System Y to fail. TIGER counts

TABLE I

Critical Equipments Computation



1. TIGER events occurring in 90 hour mission, System Y

Time	Event	System Status	Total System Downtime
0.00	Mission start	UP	0.00
8.96	Part A fails	UP	0.00
34.04	Part B fails	DOWN	0.00
61.02	Part C fails	DOWN	26.98
90.00	Mission end	DOWN	55.96

2. Breakdown of total system downtime among Critical Equipments contributing to system downtime.

Period of Total System Downtime	Number of Parts Down	System Downtime Divided Among Down Parts A	B	C
34.04-61.02 (26.98)	2 (A,B)	13.49	13.49	0.00
61.02-90.00 (28.98)	3 (A,B,C)	9.66	9.66	9.66
Total Downtime		23.15	23.15	9.66
Percent of Total System Downtime		41.37	41.37	17.26

both parts A and B as critical equipments because both contribute to system downtime. At time 34.04, system downtime begins and continues until the end of the mission at time 90.0. Thus total system downtime is

$$90.0 - 34.04 = 55.96 \text{ hours}$$

At time 61.02, Part C fails. Part C is also considered to be a critical equipment for the period from 61.02 to 90.0 (28.98 hours) even though System Y is in a down status during this period. As shown in Part 2 of Table I, TIGER divides system downtime during the period 34.04 to 61.02 (26.98 hours) between the two parts (A and B) that are in a down status. Parts A and B are each credited with 1/2 of 26.98 hours, or 13.49 hours each during this period. TIGER then divides the system downtime for the period 61.02 to 90.0 (28.98 hours) between parts A, B and C because all three parts are in a down status during this period. Parts A, B, and C are each credited with 1/3 of 28.98 hours, or 9.66 hours each during this period.

Therefore, total system downtime is divided between the three parts A, B, and C as follows: A: 23.15 hours, B: 23.15 hours, and C: 9.66 hours. These hourly total are also converted to percentages of total system downtime by part. Parts that are large contributors to system downtime can be easily identified through the Critical Equipments Summary and inventory models can then be analyzed to isolate possible weaknesses. Explanations of the other TIGER statistics can be found in the TIGER Manual [Ref. 4].

3. TIGER Subroutines

TIGER in its present form at the Naval Postgraduate School is written in FORTRAN, utilizing subroutines as major subdivisions of the program. A short summary of the purpose of each subroutine is presented below.

MAIN Program: The majority of data is input. TIGER statistics are calculated once after each mission completion and again after all missions are completed.

Subroutine PACK: Equipment configuration data and phase operating rules are input. Inventory levels are computed.

Subroutine RUN: TIGER next event calculations are done. This subroutine is called at the start of each new phase within a mission.

Subroutine TTE: Random numbers are generated to provide times for part failures or repairs. Inventory levels are monitored. Major changes to this subroutine were made for this study.

Subroutine STATUS: Equipment(s) are reviewed after each event for status (up or down) of the main system and all parts.

Subroutine STANDBY: TIGER program arrays are indexed.

Subroutine EVENT: Events (part failures, repair, etc.) are sorted to find earliest time. Major changes to this subroutine were made for this study.

Subroutine APPLE: Statistics generated during a mission are summarized.

Subroutine SPARES: This subroutine is used to input inventory levels to the main program.

Subroutine ASPARE: ASO Manual inventory levels are computed. This is a new Subroutine.

Subroutine RIMAIR: RIMAIR inventory levels are computed. This is a New Subroutine.

C. TIGER CHANGES

1. Aircraft Sortie Simulation

One of the major changes made to TIGER permitted the simulation of multiple aircraft sorties over a period of ninety days. Since TIGER was originally designed to test ship systems that underwent a few lengthy phases, variable dimensions had to be changed to allow for the many more phases that were required. With these new changes a 24-hour period may be divided up into as many as four phases. Figure 2.1 shows a sample combination of phases that can be arranged. This combination was then replicated once for each day in the mission.

Phase 1	Phase 2	Phase 3	Phase 4
← 4 →	← 8 →	← 3 →	← 9 →
Fly	Ondeck	Fly	Ondeck

Figure 2.1. Phase Sequence Combination.

During each phase a number of aircraft may be operated. Since this version of TIGER does not allow for separate aircraft (systems) to be operated in different phase sequences, aircraft were operated simultaneously. For this study three aircraft were operated in "series" operation. That is, three identical aircraft systems were operated

with the requirement that all aircraft must be in an up status for the combined trio system of aircraft to be in up status.

2. <u>Equipment: Repair and Resupply</u>

TIGER was modified so that parts that failed and were removed from the aircraft, known as carcasses, could be tracked through the repair and resupply system. The inventory algorithms studied assumed a one-for-one repair policy; for each part turned in, another is issued. Figure 2.2 shows a schematic of the overall repair and resupply pipeline. When a part fails, it has two different pipelines it can follow.

Part Failure → P (BCM) → Part Ordered Off Ship

Part Failure → 1-P (BCM) → Part Repaired on Ship

Figure 2.2. Part Pipelines.

With a probability = P (BCM), it will be considered "beyond the capability of local maintenance". In this case the part will be shipped to the depot level repair center off-ship, and a replacement part will be ordered. The time to receive a replacement part is known as the order and shipping time, OST. This time can vary depending on the stock level of the part at the depot, the location of the ship, and whether or not resupply to the carrier is possible (wartime scenario). Tiger will assign an exponentially

distributed OST, with mean equal to the SRTIM parameter of the part. SRTIM is defined as the off-ship order and shipping time of the specific part type. This time is placed in a new event time queue, RFITIM. Each part type has its own RFITIM queue that tracks all parts placed in the pipeline.

The exact number of parts in the pipeline is limited to the number of parts originally stocked. This computation is done through the NOP (number of parts) array. Whenever the NOP level equals the original inventory level, no more parts are available until a part is resupplied or repaired. The RFITIM queue is sorted to find the earliest repair time.

The failed part can also be placed in the repair pipeline, with a probability of 1-P (BCM). This corresponds to the part being repaired at the ship repair facility, the Aviation Intermediate Maintenance Department (AIMD). The part is assigned an exponentially distributed repair time, with mean equal to the REPTIM parameter of the part. REPTIM is defined as the on-ship repair time for the specific part type. This time is also placed in the RFITIM event queue. This queue runs independently of the main TIGER event chain, known as ETIME; but RFITIM does follow phase type rules outlined previously.

3. New TIGER Subroutines

Two new subroutines were introduced into TIGER. The first, ASPARE, calculates inventory levels based on ASO

Manual instructions for AVCAL determination. The second subroutine, RIMAIR, calculates inventory levels based on RIMAIR policy instructions. Both algorithms were used by Boatwright [Ref. 6]. Later chapters will examine these inventory policies.

Major changes were also made to the TTE and EVENT subroutines in order to include new repair algorithms (discussed in II.C.2), and new phase sequence rules (discussed in II.C.1). Input data cards were changed. A full listing of input card formats used in this study can be found in Appendix A. Appendices B and C contain example input data sets which are read into TIGER from separate files. A complete listing of TIGER as utilized in this study is included in Appendix D.

4. TIGER Validation

The complexity of the TIGER program makes extensive validation difficult. Two simple scenarios were chosen in order to validate this version of TIGER. Scenario One involved a single flight phase of 100 hours, with a mission time of 100 hours. Two parts, each with a MTBF = 100 hours, were arranged, first in a series configuration and then in a parallel configuration. This short mission time allows for the possibility of a successful (no failure) mission.

With this simple equipment configuration, derivation of the mathematical expression for the theoretical system availability is given in Ref. 7. The average availability can be found from the expression:

$$AVA = E(T)/MT, \tag{1}$$

where MT is the mission time = 100 hours. This assumes that no repair is possible during the 100 hour mission. E(T), the expected lifetime, is

$$E(T) = \int_0^\infty t\ f(t)\ dt. \tag{2}$$

For component analysis,

$$f(t) = d(F(t))/dt = d(1 - R(t))/dt, \tag{3}$$

where R(t) is the survivor or reliability function. For a series system the reliability is

$$R(t) = R1(t) * R2(t), \tag{4}$$

where R1(t) and R2(t) are the reliability functions for components 1 and 2. Assuming exponential failure times, Eq. (4) becomes

$$R(t) = EXP(-\lambda_1 t) * EXP(-\lambda_2 t);$$

$$= EXP(-(\lambda_1 + \lambda_2)t) = EXP(-\lambda^* t); \tag{5}$$

where $\lambda^* = \lambda_1 + \lambda_2 = 1/50$.

Substituting (5) into (3), f(t) can be expressed as

$$f(t) = \lambda^* EXP(-\lambda^* t). \tag{6}$$

Substituting (6) into (2) now gives

$$E(T) = \int_0^\infty t\lambda^* EXP(-\lambda^* t)\ dt. \tag{7}$$

The above expression assumes an infinite operating period. In our problem, time is truncated at 100 hours. Therefore if t is the system failure time, the mission lifetime is:

$$T = \begin{cases} t & \text{if } t \leq 100 \\ 100 & \text{if } t > 100 \end{cases}$$

Thus, for our case, equation (7) is modified as follows:

$$E(T) = \int_0^{100} t\lambda^* EXP(-\lambda^* t)\ dt + 100 \int_{100}^{\infty} \lambda^* EXP(-\lambda^* t)\ dt$$

$$= 29.78 + 13.53 = 43.31.$$

Finally,

$$AVA\ (\text{series}) = 1/100\ (43.31) = \boxed{0.4331.}$$

For a parallel configuration system, its reliability, R(t), can be expressed as follows:

$$R(t) = R1(t) + R2(t) - R1(t) * R2(t). \qquad (8)$$

The last term in (8), R1(t) * R2(t), is the series reliability term (43.31) computed above. Also note that for this problem R1(t) equals R2(t) because the two components have identical MTBFs. Substituting 43.31 for the last term in (8), E(t) is computed as in the series system above to be

$$= 2 * \left[\int_0^{100} t\lambda_1 EXP(-\lambda_1 t)\ dt + 100 \int_{100}^{\infty} \lambda_1 EXP(-\lambda_1 t)\ dt \right] - 43.31$$

$$= 2 * (63.21) - 43.31 = 83.11.$$

This gives:

$$AVA\ (\text{parallel}) = (83.11)/100 = 0.8311.$$

Scenario Two involved a single flight phase of 5000 hours, utilizing the same two systems. A phase of 5000 hours ensures the failure of the system and an estimate can only be made of the expected lifetime of the system. For the series system, the expected lifetime is

$$E(t) = \int_0^{5000} t\lambda^* EXP(-\lambda^* t)\ dt + 5000 \int_{5000}^{\infty} \lambda^* EXP(-\lambda^* t)\ dt.$$

Therefore,

$$E(t) = 50.0 + 0.0 = 50.0 \text{ hours.}$$

For the parallel system, the expected lifetime is

$$E(t) = \int_0^{\infty} t(2\lambda_1 EXP(-\lambda_1 t) + \lambda^* EXP(-\lambda^* t)\ dt$$

$$= 2 * (100.0) - 50.0 = 150.0 \text{ hours.}$$

Results for validation runs are shown for both scenarios in Table II. One thousand iterations were done for each run.

TABLE II

TIGER Validation Results

Scenario 1 (100 Hours)

Run	Seed	AVA (series)	AVA (parallel)
1	2222	0.4294	0.8269
2	1245	0.4360	0.8444
3	1357	0.4341	0.8453
Theoretical Value		0.4331	0.8331

Scenario 2 (5000 Hours)

Run	Seed	E (Lifetime Series)	E (Lifetime Parallel)
4	2222	50.3	149.8
5	1245	50.7	153.4
6	1357	49.5	144.9
Theoretical Value		50.0	150.0

III. ASO MODEL

A. MODEL DESCRIPTION

The Navy Aviation Supply Office Manual model for determining the AVCAL is based on the repair/resupply pipeline displayed in Figure 3.1. Failure of parts in a ninety-day period create a demand, QTRDEM. With a probability equal to P, parts are beyond the capability of shipboard maintenance (BCM), or with a probability 1-P are determined to be repairable onboard ship.

The BCM'ed parts are sent off-ship, either to be disposed of or to be repaired at the depot repair maintenance facility ashore. In either case a replacement part is ordered through the requisition pipeline. The order and shipping time (OST) is the time from order until receipt of a new part. A part repairable at the shipboard level experiences a delay in the repair pipeline called the turn around time (TAT). The average number of parts in this pipeline is the mean repair pipeline (MRP). When parts are received from either pipeline they are placed back into the local (retail) inventory.

The following assumptions are made in the model [Ref. 8]:

Demand is a Poisson process.

Demand rates are stationary over time (no surge or cyclic demand rates).

OST and TAT are independent of demand.

The repair pipeline is never saturated.

Items are requisitioned on a one-for-one basis (S-1, S ordering policy).

All demands are satisfied by either immediate replacement from supply, shipboard repair, or requisition (back order).

Part cannibalization does not occur.

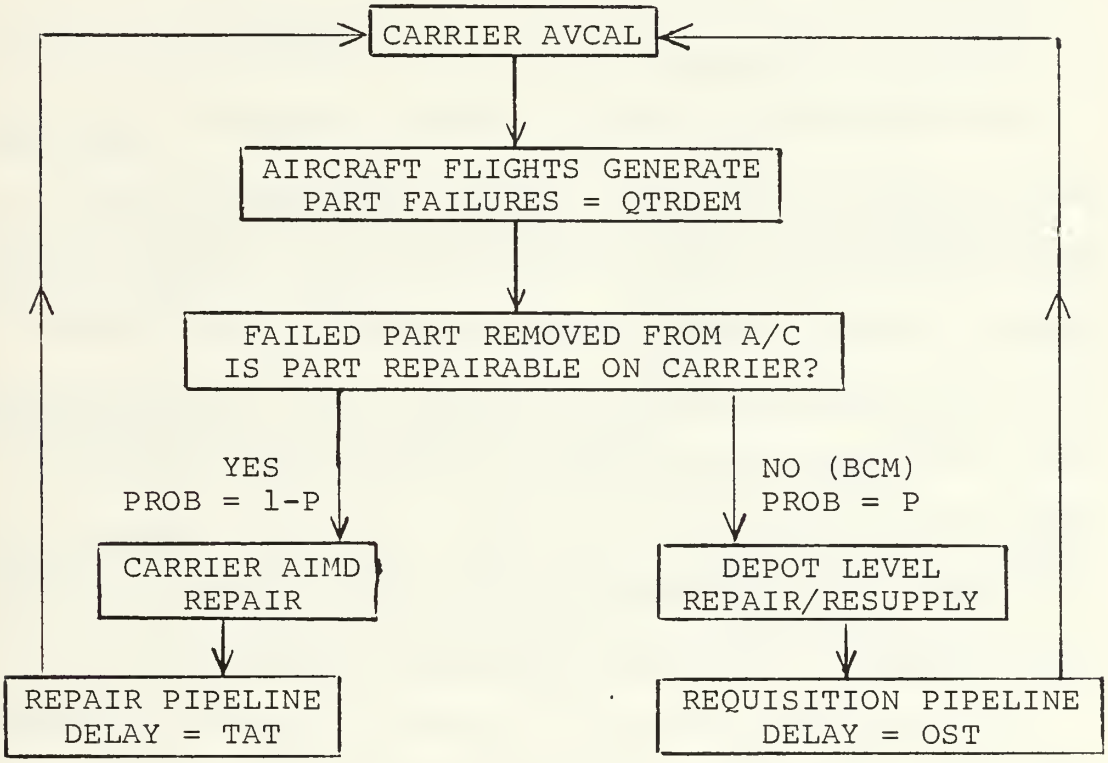


Figure 3.1. ASO Repair/Resupply Model.

These assumptions are very generous, but the net affect is that this model is a fairly simple one. The first assumption leads to demand over a given period t being distributed as Poisson, with mean = (QTRDEM * t). It can be shown [Ref. 9] that the two "pipelines", ship repair and off-ship requisition, are independent Poisson processes with means (P * QTRDEM)* t and (1-P)*QTRDEM*t, respectively. The number of items in the ship repair pipeline is also Poisson, with mean RPQ.

B. INVENTORY DETERMINATION

The Navy Aviation Supply Office's (ASO) Provisioning Manual [Ref. 10] furnishes policy and procedures for determining AVCAL range and depth levels. Although the process of generating a complete AVCAL is quite complex, the basic guidelines used for repairables are concise. Navy 3M data, contractor usage data, and laboratory results are combined to predict failure rates. The ASO Manual refers to failure rate prediction as "the most important function of provisioning" in the determination of the AVCAL.

The Outfitting Directive, issued by the Type Commander, specifies the Type/Series/Model of aircraft to be supported, the number of aircraft, and the number of flight hours per month per aircraft. The Allowance Requirement Registers (ARRs), which make up the AVCAL, are divided into three major parts:

Part I Attrition Support

Part II Rotatable Pool Items

Part III Special Support Requirements

This study will deal only with the first two parts. The following data elements are used to construct ARR's:

Maintenance Cycle (MC): Normally 100 hours for aircraft and installed equipment.

Units per Component/Aircraft (UPA): Number of parts of type X installed on each aircraft. An individual UPA exists for each part type X.

Planned Operating Hours: Planned aircraft utilization per month in hours.

Number of Aircraft: Number of aircraft supported by this ARR.

Maintenance Replacement Factor (MRF): For repairable items, the number of times that an item will be BCM at organizational (squadron) and Intermediate (AIMD) levels during one MC.

$$MRF = \# \text{ BCM's}/(MC * UPA)$$

Rotatable Pool Factor (RPF): Predicted number of removal/IMA repair cycles in one MC.

$$RPF = (\text{Predicted } \# \text{ of repairs})/(MC * UPA)$$

Turn Around Time (TAT): Average number of days between removal of a repairable item for processing at the AIMD and return to Ready For Issue (RFI) condition. This estimate includes time to schedule, fault isolate, disassemble, repair, assemble, and test a repairable assembly.

The candidates for inclusion in the AVCAL are chosen as follows. The attrition quantity in any ninety day period is determined as follows:

(1) Compute Flight Hour Factor (FHF) for aircraft:

$$FHF = (\text{avg. } \# \text{ of Aircraft}) * (\text{Operating Hrs./Qtr.})$$

(2) Compute Expected number of Maintenance Cycles per quarter:

$$\# \text{ of MC} = FHF/100$$

(3) Compute Attrition Quantity (D):

$$D = MRF * UPA * (\# \text{ of } MC)$$

1. Attrition Rules

Attrition items are stocked to replace those parts that are BCM'ed at the organizational or intermediate level. The range rules for attrition parts depend on whether the part is also included in the rotatable pool quantity. If a part is supported in the rotatable pool, it must have a demand (D) greater than or equal to one per quarter to be eligible for the attrition portion of the AVCAL.

If the part is not supported in the rotatable pool, the range rules for attrition are different. These low demand items may still qualify for the attrition allowance under the following guidelines:

a. Items with a unit cost of $5000 or more will qualify if the predicted demand is equal to or greater than one in a six month period. This equates to an attrition quantity of at least 0.50 before any units will be carried in the AVCAL.

b. Items with a unit cost of less than $5000 will qualify if the predicted demand is equal to or greater than one in a nine month period. This equates to an attrition quantity of at least 0.34 before a quantity of one will be carried in the AVCAL.

Attrition quantities are used to determine attrition range candidates as noted above. Once a candidate has been

selected, the depth or amount to be stocked is computed by rounding the attrition quantity to the nearest integer. A minimum of one is stocked for all range qualifying candidates.

2. Rotatable Pool Rules

The rotatable pool portion of the AVCAL allowance was intended to support the fast moving, critical parts and assemblies required to support the aircraft. These items must be capable of repair at the intermediate level (AIMD). The raw pool quantity (RPQ) is the average number of units repaired by the AIMD in a 90 day period and is given by:

$$RPQ = (RPF * TAT * UPA * (\# \text{ of MC/in 90 days}))/90$$

It should be noted that the value for TAT is an averaged value that is truncated to a maximum of twenty days. This data element will be further discussed later in the chapter. Using RPQ as the mean, the Poisson distribution is used to find the depth which will provide 90% protection against being short at least one unit in ninety days. This depth, called the rotatable pool allowance (RPA), provides 90% protection for those parts which have carcasses tied up in the repair pipeline. Therefore,

$$P(X \leq RPA) = 0.9 ,$$

where X = # of units of a part being repaired in the shipboard pipeline. Under the assumption that X is Poisson distributed,

$$P(X \leq RPA) = \sum_{X=0}^{RPA} \frac{EXP(-RPQ) * (RPQ)^X}{X!}$$

Using these calculations, an RPQ of 0.11 is the minimum value that will require an RPA of one. Below the 0.11 level a stock quantity of zero satisfies the 90% protection, and no part is stocked for the pool.

C. MODEL LIMITATIONS

The ASO model is the oldest of the three models discussed in this thesis. It is the only model developed before data automation and powerful computers became widespread in the Navy. This partially explains the model's simple approach to the inventory problem. Procedures are simple enough that inventory levels could be calculated by hand for each part, one at a time, with the use of one short table from the ASO Manual. One noteworthy weakness in the model is the omission of the concept of budget. The only direct reference to dollar amounts is in the use of $5000 as a cutoff amount for attrition allowance. But even this figure has become completely arbitrary because there is no provision for its change or update and because it applies to inventories with parts typically ranging in price from a few hundred dollars to over several hundred thousand dollars.

Mitchell [Ref. 11] pointed out that limiting TAT to twenty days is not a true reflection of the real repair pipeline operation. A breakdown of the TAT elements is shown below in Figure 3.2. The limit values were developed at ASO in a study conducted in 1977 [Ref. 12]. The limit values tend to understate the problems encountered in the repair pipeline. The

values are applied across all parts, although the complex equipments encounter longer times than the simple parts.

	TAT element	Limit (days)
IP:	In-process time	1
SKD:	Scheduling time	3
RPR:	Repair time	8
AWP:	Awaiting parts time	20
TAT:	Total time	20

Figure 3.2. TAT Elements.

The ASO model is tasked to achieve material availability goals and stockage criteria promulgated in OPNAVINST 4441.12A [Ref. 13]. For ships, the objective for overall AVCAL performance is to fill 75% of all demands and to provide overall availability of 85% for items stocked. But as noted in the Navy Fleet Materials Support Office RIM-AIR Study [Ref. 14], the ASO model has historically failed to do this. Fleet aircraft availability is often achieved only through a constant process of selective cannibalization of squadron aircraft parts. For example, in an E-2C squadron with four aircraft aboard a carrier, one aircraft is designated the "parts locker" in order to overcome shortcomings in both the repair and requisition pipelines.

There is a disadvantage in the ASO criteria that attrition and repair demand be segregated. Separate range criteria are applied to determine attrition and repair pool support. This splitting of demand results in non-stockage of items that would have been stocked had demand been combined. This contributes to the overall conservative approach that characterizes ASO Manual AVCAL levels.

IV. RIMAIR MODEL

A. MODEL DESCRIPTION

During the Seventies all DOD budget policies came under close scrutiny by civilian government leaders. The DOD Retail Inventory Management and Stockage Policy (RIMSTOP) Study was issued in 1976 to set guidelines for retail level inventory support provided by the military services [Ref. 14]. Out of RIMSTOP originated DOD Directive 4140.44 (Supply Management of the Intermediate and Consumer Levels of Inventory), and DOD Instructions 4140.45 (consumable items), 4140.46 (repairable items) and 4140.47 (war reserves). DODI 4140.46 [Ref. 15] dictates that:

> "the following levels will be computed for each repairable item to be stocked at the intermediate level on a demand-supported basis:
>
> (1) Repair Cycle Level (RCL). The RCL is a function of the anticipated number of maintenance replacements that will be repaired locally and the item's repair cycle time.
>
> (2) Order and Shipping Time Level (OSTL). The OSTL is a function of the anticipated number of maintenance replacements that will require supply from external sources and the item's order and shipping time.
>
> (3) Safety Level (SL). The SL is a function of the capabilities that the repair cycle time will be exceeded, the order and shipping time will be exceeded, the maintenance replacement rate will be higher than forecasted, and a number of maintenance replacements, anticipated for repair at the activity, will require resupply from external sources.

(4) Operating Level (OL). The OL is an Economic Order Quantity (EOQ) and is a function of the cost to order and the cost to hold an item of inventory.

(5) Replenishment. Replenishment action will be taken when the asset position reaches the reorder point."

In addition, DODI 4140.47 (Secondary Item War Reserve Requirements Development) authorizes increments to the order and ship time, repair cycle and safety levels to satisfy wartime recurring demands over and above the peacetime demands. An additional Resupply Delay Time (RDT) level is also authorized to provide material coverage of anticipated delays in the wartime retail level supply pipeline.

Commander, Naval Supply Systems Command (COMNAVSUPSYSCOM) proposed a pipeline model that would adhere to these DOD policies while attaining Navy availability goals. This model was designated Repairables Integrated Model for Aviation (RIMAIR). In addition to the levels mentioned above, RIMAIR added a level of stock that assures a self-supporting capability for a prescribed period of time, known as an "endurance delta". The same assumptions stated for the ASO model apply to this model.

RIMAIR produces a total depth of stock that equals:

$$OL + RCL_w + MAX \begin{Bmatrix} OST_p + EDT \\ \\ OST_w + RDT \end{Bmatrix} + SL,$$

where

OL = operating level;

RCL_w = repair cycle level computed with a wartime flying hour program;

OST_p = order and ship time level computed with a peacetime flying hour program;

EDP = endurance period support level to assure self-supporting capability to satisfy wartime demands for a prescribed period of time;

OST_w = order and ship time level computed with a wartime flying hour program;

RDT = resupply delay time level;

SL = total safety level based on the sum of RCL and the MAX computation.

The peacetime operating stock (POS) levels may be separated from the total depth:

$$POS = OL + RCL_p + OST_p + SL_p$$

The endurance delta represents the difference between OST_p + EDP and OST_w + RDT [Ref. 14].

B. INVENTORY DETERMINATION

1. Steady-state Supply Effectiveness

Appendices C and D of the FMSO RIM-AIR Study [Ref. 14] provide the mathematical background for this model. Initially assume no stock is carried. The repair and requisition processes can be modeled mathematically as stochastic queuing processes in which non-RFI (failed) units arrive, wait for a RFI replacement then leave. The average number of items in a queuing process is given by the following relationship:

$$L = \lambda * W$$

where L = average number of units in process

λ = average arrival rate

W = average waiting time in process

The number of requirements for a RFI replacement in the requisition process is the requisition pipeline. The number of non-RFI units in the repair process is called the repair pipeline. Given the above relationship, the average number of non-RFI units in the repair and requisition pipelines may be expressed as follows:

$$\begin{aligned} L_T &= L_{REP} + L_{REQ} \\ &= \lambda_{REP} * W_{REP} + \lambda_{REQ} * W_{REQ} \\ &= \frac{RPF * MC_{90}}{90} * TAT + \frac{MRF * MC_{90}}{90} * OST \end{aligned}$$

where

L_T = total non-RFI units waiting for replacement

L_{REP} = non-RFI units in the repair process

L_{REQ} = non-RFI units in the requisition process

λ_{REP} = arrival rate for repair proces

λ_{REQ} = arrival rate for requisition process

W_{REP} = waiting time for repair process

W_{REQ} = waiting time for requisition process

MC_{90} = waiting cycle program for 90 days

The actual number of units in the repair and requisition pipelines at some point in time is a random variable. The following assumptions are made in order to postulate a probability function for this random variable:

The arrival process in Poisson.

The repair times have a distribution which is independent of the arrival process.

The arrival rates and services rates are stationary over time.

Arrivals are always single units.

Every arrival enters either the repair or requisition process and completes service before departing.

Given these assumptions, the number of units N in the repair and requisition pipelines will be Poisson distributed with mean L_T for repairables. That is, the probability that N = n, is found from the expression:

$$P(N=n) = EXP(-L_T) * (L_T)^n/(n!).$$

The probability that there are no backorders, called the protection, is computed as follows:

$$\text{Protection} = \sum_{n=0}^{S} P(N=n) \quad ,$$

where S = stock quantity.

When the number of units in the repair or requisition processes is strictly less than the stock quantity, there is at least one RFI unit available in stock to satisfy a demand should one occur. Since demands are assumed to always be for one unit, only one unit needs to be in stock when a demand occurs in order to satisfy that demand. The probability of satisfying a demand is called the fill rate (FR) and is computed as follows:

$$FR = \sum_{n=0}^{S-1} P(n)$$

The expected number of satisfied demands is found by multiplying the fill rate by the expected number of demands. The expected demands (D) for a 90 day period is computed as follows:

$$D = (MRF + RPF) * MC_{90}$$

Thus, the expected gross supply effectiveness, which is the percentage of demands satisfied immediately from stock, can be computed as follows:

Expected Supply Effectiveness =

$$\frac{\sum_{i=1}^{m} FR_i * D_i}{\sum_{j=1}^{Q} D_j} \qquad (1)$$

where

m = number of stocked items
Q = number of installed items
i = index of stocked items
j = index of installed items

Expected net supply effectiveness is obtained by summing expected demand over stocked items in the denominator.

2. Optimization

The objective of the optimization of this model is to find an inventory that gives the maximum possible effectiveness for a given cost. The effectiveness measure used is the expected gross supply effectiveness derived above. Expected units demanded for installed items remain constant.

The optimization maximizes expected supply effectiveness. The problem may be stated as follows:

$$\text{Maximize} \sum_{i=1}^{m} E_i * D_i * FR_i$$

$$\text{subject to} \sum_{i=1}^{m} C_i * S_i = B$$

where

i = item index;
E_i = Item essentiality code;
D_i = Expected demand;
FR_i = Fill rate per item;
C_i = Unit price;
S_i = Stock Quantity;
B = Cost target.

RIMAIR uses the method of Lagrange multipliers to solve this problem. Formulating the Lagrangian function from the problem above gives:

$$L(\lambda, \overline{S}) = \sum_{i=1}^{m} E_i * D_i * FR_i - \lambda * \left(\sum_{i=1}^{m} C_i * S_i - B\right)$$

Because of the discrete nature of the demand distribution, the stockage levels are determined using finite differences. Observe that $L(\lambda, \overline{S})$ is separable in the items. Thus the Lagrange function can be written as:

$$L(\lambda, \overline{S}) = \sum_{i=1}^{m} L_i(S_i; \lambda) + B \lambda$$

where $L_i(S_i; \lambda) = E_i * D_i * FR_i(S_i) - \lambda C_i S_i$.

For a given value of λ the stockage level for item i is then the largest integer S_i such that:

$$\Delta L_i(S_i; \lambda) = L_i(S_i+1; \lambda) - L_i(S_i; \lambda) > 0$$

This is found to be the largest integer S such that

$$p_i(S_i) > \frac{\lambda * C_i}{E_i * D_i} \qquad (2)$$

where $p_i(S_i)$ is the probability density function of the Poisson pipeline distribution, given earlier as $P(N=n)$.

The "optimal" stockage level corresponds to the solution to this equation when $\lambda = \lambda^*$ where λ^* is that value such that $\sum C_i * S_i = B$. (Due to the discrete nature of the items the required budget may never exactly equal B and consequently, the Lagrange solution may not be optimal. It will produce, however, an undominated solution for each budget amount actually consumed.)

The procedure outlined above for finding $\bar{S}^*$ can be applied with any value of λ. When used with λ^*, it produces the solution to the original problem. When used with any other λ, it produces an inventory that still maximizes the Lagrangian function with respect to $\bar{S}$ but does not satisfy the budget constraint. The process then becomes one of finding the correct λ, until the cost is close to the target B.

The RIMAIR implementation of the solution procedure as applied to the stockage level for the ith item is summarized

below:

a. Select the Lagrange multiplier.

b. Find the largest integer which is less than or equal to L_T as an initial value for S_i.

c. If

$$p(S_i) < \frac{\lambda * C_i}{E_i * D_i}$$

do not stock the item. This situation is depicted in Figure 4.1, Case A. The Poisson density function $p(S_i)$ is everywhere less than the value for $\lambda * C_i/E_i * D_i$. Therefore the optimal stockage level equals zero. If

$$p(S_i) > \lambda * C_i/E_i * D_i$$

go on to step d. The second situation is shown in Figure 4.1, Case B. S_i is initially set equal to 2 (the largest integer less than or equal to the mean $L_T = 2.5$).

d. Increment S_i by one.

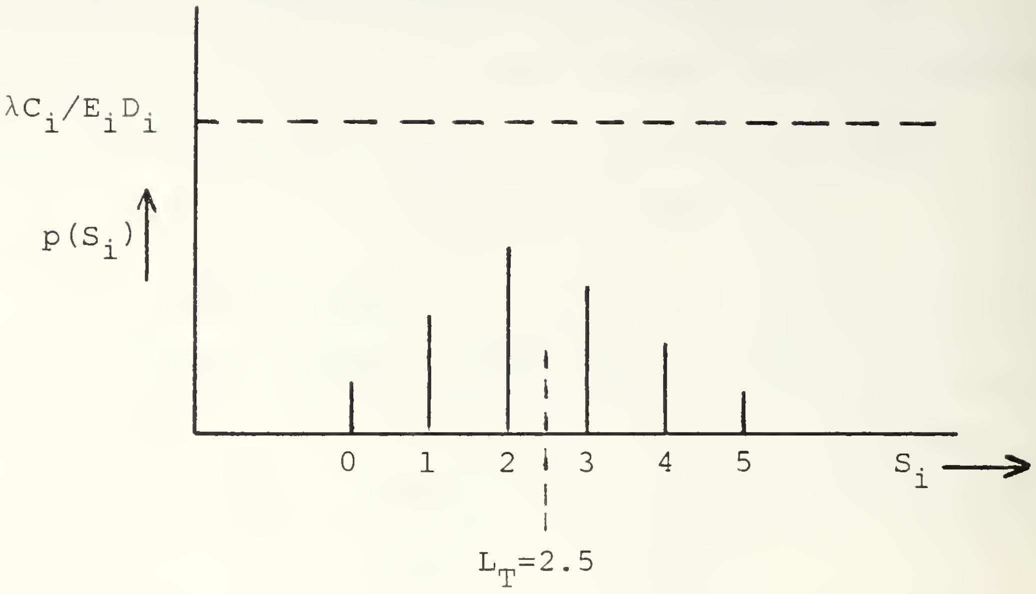
e. If

$$p(S_i) < \frac{\lambda * C_i}{E_i * D_i}$$

select S_i as S_i^* and stop; otherwise go to step d. In Case B, S_i=4 would be chosen as S_i^*. (Note that this implementation will select S_i^* to be one larger than that which would be generated by Equation 2.)

f. Compare the optimal stockage level to external constraints and adjust accordingly.

Case A: Zero Stockage Level



Case B: Positive Stockage Level

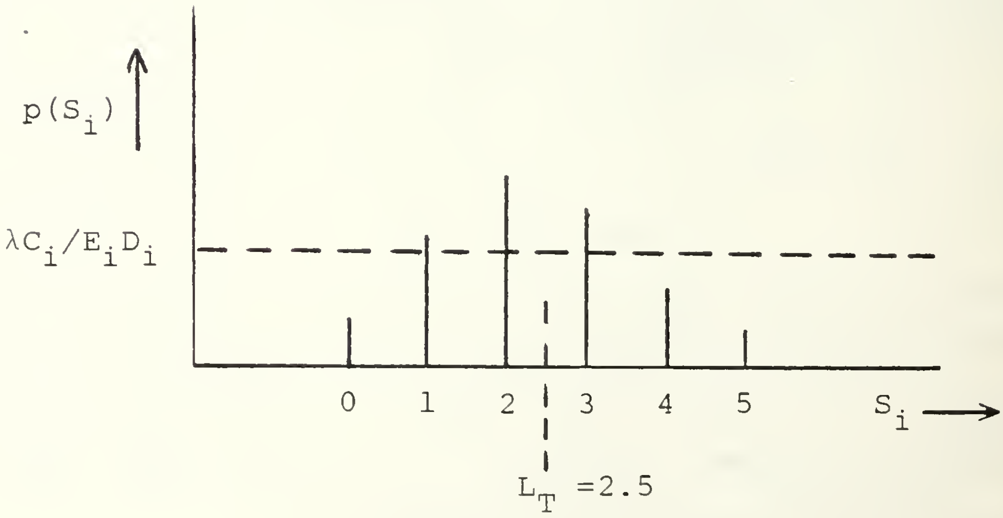


Figure 4.1. RIMAIR Stockage Level Options.

g. Iterate through all items, and compare the final total inventory cost with the budget target B. If the total cost is not within preassigned limits return to step a. For example, if total cost is not within plus or minus 1% of target budget, begin at step a with a new lambda value and try to get total cost within 1% limits.

This procedure simultaneously produces the range and depth criteria. That is, if the optimal stockage level is greater than zero, then the item is stocked.

3. External Constraints

Step (f) includes comparing stockage levels to constraints external to the original Lagrange problem. The maximum constraint is the sum of a ninety-nine percent protection on the mean basic pipeline (BP) and the operating level. BP is defined as:

$$BP = L_T + \text{ENDURANCE LEVEL} \tag{3}$$

Since the basic pipeline quantity is assumed to be Poisson distributed with mean BP, the 0.99 protection level would be the smallest quantity S such that:

$$\sum_{X=0}^{S} \text{EXP}(-BP) * (BP)^X/X! \geq 0.99 \tag{4}$$

The operating level is computed as follows:

$$OL = \sqrt{(2 * A * Y)/(IC)} \tag{5}$$

Where: Y= annual demands
C= unit price
2A/I= constant (approximately 559)

The maximum constraint is the sum of S from (4) and OL from (5) above.

The minimum constraint is the sum of (3) and (5) above.

C. MODEL LIMITATIONS

The RIMAIR model corrected several of the deficiencies of the ASO model. It was recognized that the ASO model's attrition allowance, which was theoretically provided to support wartime mobilization operations with resupply delayed or cut off, was in fact supporting the number of items in the wholesale resupply pipeline during normal operations. The RIMAIR model includes the addition of stock to the attrition portion of the allowance to support the expected order and shipping time experienced during peacetime, and the addition of a wholesale resupply pipeline to the repair cycle pipeline for the purpose of providing Poisson protection to the entire pipeline.

One of the potential strengths of the RIMAIR model is the inclusion of the item essentiality code parameter E_i. This code was developed to reflect the relative importance of parts to total system availability. Suggestions for use of this parameter are addressed in Boatwright [Ref. 6]. An ideal code would be influenced by the part's MTBF, by the system configuration (whether or not the part had backups), and by the role that the part played in contributing to aircraft mission completion.

Incorporation of these concepts into the essentiality code is difficult. For this study an item essentiality code equal to one was assumed for all parts. One reason for this was because all parts were considered equally essential for aircraft mission performance.

The Lagrange multiplier provided control for budget levels as discussed above. By decreasing lambda the inventory cost would increase, or by increasing lambda the inventory cost would decrease. This budget control function was discrete. The actual inventory cost could vary from the target budget by as much as the cost of a single part.

The RIMAIR algorithm was included within the TIGER program as a separate subroutine. First, RIMAIR inventory levels are computed in the subroutine and second, TIGER simulates aircraft flights with these RIMAIR stocks as input. Lambda values are included in the input data file, external to the TIGER program. Lambda values are changed and new budget levels are then examined to see if they meet the target budget.

The RIMAIR and ASO models share some of the same weaknesses because they are based on the same underlying assumptions. The problem that the ASO model encountered with TAT, discussed in III.C, is also present in RIMAIR. This points out that there are problems in the inventory decision process that exist above the model level. In this case, it is with the Navy process of data collection of TAT.

Another comparison can be made between the two models as far as workload required to support it. The RIMAIR model increases the workload compared to the ASO model because RIMAIR introduces two new parameters, the item essentiality code and the lambda value. The problem with the essentiality code, as mentioned above is how to assign it; faulty coding can result in unbalanced AVCALs. Time must be spent assigning and updating these codes. Since the lambda value is assigned external to the RIMAIR subroutine, time is spent checking budget levels and resetting lambda values. One improvement to the present algorithm would be to include a loop in the program that would change the lambda value depending on proximity to target budget.

Both the ASO and RIMAIR models are retail level, single echelon models. This means that they calculate AVCALs only for the organizational level facility. Multi-echelon models have been developed that spell out stock levels at organizational, intermediate and depot level facilities. The next chapter will examine one of these multi-echelon models, ACIM, that can also be used for the single-echelon case.

V. ACIM MODEL

A. MODEL DESCRIPTION

The Naval Sea Systems Command's Availability Inventory Model (ACIM) was developed after the Chief of Naval Operations directed that "a sophisticated availability-based sparing technique be developed and applied on a selected basis for equipments which require a level of readiness above that which standard policies can provide [Ref. 16]."

In response to this CNO direction, the Chief of Naval Material issued NAVMATINST 3000.2. This instruction established Operational Availability (A_o) as the primary measure of material readiness for Navy weapons systems and established policy for A_o analytical techniques. Subsequently CHNAVMAT recommended, and CNO approved, a standard availability centered optimization model for use by all program managers in determining consumer level stockage quantities for selected equipments. This ACIM model develops repair parts allowances to achieve a specified A_o at the minimum possible inventory cost.

This thesis will investigate ACIM model version 2.0, developed by CACI-Inc Federal and implemented by Henry J. Watras for use on the NPS IBM 3033. This chapter will describe the ACIM model as it applies to AVCAL determination

in this thesis. A more detailed analysis of this model can be found in McDonnell [Ref. 17] and in the ACIM Handbook [Ref. 16].

The underlying assumptions of the ACIM model are listed below.

1. Included parts are organized in terms of an equipment with topdown breakdown. Multiple units of a part within a given next higher assembly are represented only once in the breakdown. However, if the same part appears in different locations in the structure, each appearance is treated as a unique item in the operation of the model.

2. External demands upon supply are stationary and compound-Poisson distributed.

3. All stockage locations use a continuous review, (S-1,S) ordering policy.

4. Mean times to repair are defined as constants which include all equipment repair related down times that are not supply related.

5. Component failures are independent.

6. No further demands for parts can occur when one or more parts are in down status. That is, when a part fails the system does not operate again until the failed part is replaced.

A top-down breakdown is one which starts with the highest level unit, in this case the E-2C aircraft. The next level down is the WRA level, which are the individual parts discussed in this study. Below the WRA level is the Shop Replaceable Assembly (SRA) level, the sub-SRA level, and on down until the smallest diode or resistor has been itemized. This multi-level approach is also called a multi-identured approach. For this study only WRA level

inventories will be computed although ACIM can compute stocks down to the lowest level.

The ACIM definition of availability is the same as that used in the TIGER simulation model; namely,

$$A_o = \frac{\text{UPTIME}}{\text{UPTIME + DOWNTIME}}$$

ACIM replaces uptime by MTBF and downtime by Mean Time To Repair (MTTR) plus Mean Supply Response Time (MSRT). So, A_o can be reexpressed as:

$$A_o = \frac{\text{MTBF}}{\text{MTBF + MTTR + MSRT}}$$

The MTTR and MTBF parameters are inputs to the ACIM model. The MSRT factor depends on the stockage levels and ACIM uses this dependency to achieve a target value of A_o. ACIM actually attempts to minimize MSRT in order to maximize A_o.

B. INVENTORY DETERMINATION

1. ACIM Solution Equations

The model is defined recursively by considering an arbitrary item in the system and an arbitrary facility. The system is the aircraft, and the items are the individual parts (WRAs). The structure of the model is given by the following set of definitions and equations:

a. Let i be an arbitrary item in equipment e (which may be e itself). Let u = 0 represent an arbitrary facility in the support system.

b. $M_{iu} = DEL_{iu} + R_{iu}$,

M_{iu} = mean time to return a failed item i at location u to a serviceable condition;

DEL_{iu} = expected delay per demand for item i at location u, experienced in the repair and requisition pipelines;

R_{iu} = mean time to repair item i at user location u (for on-equipment repair);

= 0 if location u does not operate the equipment.

c. $$DEL_{iu} = \frac{1}{Y_{iu}} \sum_{X \geq S_{iu}} (X-S_{iu}) * p(X; Y_{iu} * T_{iu}),$$

where S_{iu} = stock level of item i at location;

Y_{iu} = expected number of demands upon inventory for item i at location u;

$p(X; Y_{iu}*T_{iu})$ = probability of X units of stock reduction for item i at location u; the distribution may be Poisson, Normal or Negative Binomial, depending on the mean and variance of the part;

T_{iu} = mean resupply time (time to replace an inventory loss) for item i at location u.

d. $$T_{iu} = P_{iu}(L_{iu} + L'_{iu}) + (1-P_{iu}) * (R_{iu} + R'_{iu}),$$

where P_{iu} = probability that a demand for item i upon inventory at location u results in a loss (discard or sent elsewhere for repair) which must be replaced through resupply;

L_{iu} = average resupply lead time assuming stock is available at the resupply source;

L'_{iu} = additional resupply lead time due to expected shortages at the resupply source;

R_{iu} = average shop repair cycle time assuming availability of spares for items within i at the next lower indenture level;

R'_{iu} = additional shop repair cycle time due to expected shortages of spares for items within i at the next lower indenture level.

The values of L_{iu}, R_{iu}, T_{iu}, and Y_{iu} are inputs to the model.

$$\text{e.}\quad L' = \begin{cases} D_{iv} & \text{for } u = 0 \\ D_{io} & \text{for } u = 1,2,\ldots,U; \end{cases}$$

where v is the resupply source for location 0 and v=0 if the location 0 has no resupply source.

$$\text{f.}\quad R'_{iu} = \sum_{j\varepsilon i} Y_{ji} \; M_{ju} / \sum_{j\varepsilon i} Y_{ju} \,,$$

where j identifies items within i at the next lower indenture level; j = 0 if i has no subordinate parts.

$$\text{g.}\quad A_{eu} = 1/(1 + Y_{eu} * M_{eu}) \,,$$

where A_{eu} = fraction of time equipment e is available for use at location u (defined only for locations u which operate the equipment).

2. Objective Function

The overall objective of ACIM is to determine stockages levels for all items and all stockage facilities so that the expected operational availability of the equipment is maximized for a given inventory budget or, conversely, to find levels which achieve a given operational availability at least cost. This objective can be explicitly stated as follows:

Find values for S_k for all items k and locations v in the

support system which minimize DEL = DEL_{eu} for all user locations u subject to:

$$\sum_{k,v} c_k S_k \leq B,$$

where c_k = unit cost of item k;

B = given budget for spares procurement.

A similar statement can be written for the converse objective of achieving a given value for A_{eu} at least cost.

The ACIM solution to the problem defined above is found by a recursive procedure based upon equations b-g. First, however, a subproblem is defined and a solution procedure is given for the subproblem. A recursive application of the subproblem is then used to solve the original problem.

The subproblem is set up as follows. Substituting equation d in c, the expected delay per demand is given by

$$DEL_{iv} = DEL\ (S_{iv},\ L'_{iv},\ R'_{iv})$$

where the stock level S_{iv}, additional resupply time L'_{iv}, and additional repair cycle time, R'_{iv}, are considered as decision variables for an arbitrary item $i \varepsilon e$ and arbitrary location v in the support system. Suppose that values for S_i are given for all items and locations v. The subproblem is to find a particular item and location such that a one unit increase in its stock level will yield the largest decrease in DEL_{eu} per dollar investment for some user location u.

The solution of this subproblem is based upon a recognition that the family of functions D_{iv} are hierarchically related (by equations e and f), each is a function of three decision variables, and functions at the bottom of the hierarchy depend only upon the stock levels, S_{iv}.

Therefore, a marginal analysis solution procedure can be applied as follows:

Define

$$\Delta_s D_{iv} = D(S_{iv}, L'_w, R'_{iv}) - D(S_{iv} + 1, L'_{iv}, R'_{iv}) ;$$

$$\Delta_L D_{iv} = D(S_{iv}, L'_{iv}, R'_{iv}) - D(S_{iv}, L'_{iv}{}^* , R'_{iv}) ;$$

$$\Delta_R D_{iv} = D(S_{iv}, L'_{iv}, R'_{iv}) - D(S_{iv}, L'_{iv}, R'_{iv}{}^*) ;$$

where

$L'_{iv}{}^*$ = least value of L'_{iv} obtainable by a unit increase increase in stock of some part $w \varepsilon i$ at the supply source for v;

$R'_{iv}{}^*$ = least value of R'_{iv} obtainable by a unit increase in stock of some part $r \varepsilon i$ at location v.

Letting w^* represent the part which satisfies $L'_{iv}{}^*$ and $r_{iv}{}^*$ the part which satisfies $R'_{iv}{}^*$, find the largest of

(a) $\Delta_s D_{iv}/c_i$, (b) $\Delta_L D_{iv}/c_w{}^*$, and (c) $\Delta_R D_{iv}/c_{r^*}$;

and let

$$D_{iv}{}^* = D(S_{iv} + 1, L'_{iv}, R'_{iv}) ;$$

$$= D(S_{iv}, L'_{iv}{}^*, R'_{iv}) ;$$

$$= D(S_{iv}, L'_{iv}, R'_{iv}{}^*);$$

according to which of (a), (b), or (c) is largest, respectively.

With the above definitions and using equations b, e, and f, a recursion is given by:

$$L'_{iv}{}^{*} = D_{iz}{}^{*} \text{ (z = supply source for v);}$$

$$R'_{iv}{}^{*} = \left(\sum_{j\varepsilon i} Y_{jv}M_{jv} + Y_{jv} \; M^{*}_{jv} \right) / \sum_{j\varepsilon i} Y_{jv} \; ;$$

$$M_{jv} = D_{jv}{}^{*} + R_{jv} \; ;$$

where j identifies parts within i at the next lower identure, and j' = r* or contains r* as a lower level part. The recursion is initiated for items i and the location v where L'_{iv} and R'_{iv} are both zero and hence $D_{iv}{}^{*} = D(S_{iv} + 1)$.

Justification that this procedure solves the subproblem follows from convexity properties of the functions D_{iv}. The solution to the original problem is given by repeated application of the subproblem.

If the solution to the subproblem results in the availability target or budget target being met or exceeded, the original problem is solved. Otherwise a new subproblem is solved to find the next item which will result in the largest decrease in DEL per dollar investment. This process continues to build up the stockage levels until one of the two targets has been achieved.

Unlike the RIMAIR and ASO models, which were subroutines of TIGER, ACIM was run as a separate program using batch processing. The ACIM program is made up of three

subprograms that operate in sequence. The first program (Preprocessor) calculates stockage levels according to designated comparison policies. The second (Main) program of the model calculates levels according to ACIM. Stockage levels calculated by the first and second programs are passed to the third program (Postprocessor) which produces three output reports: a cost-effectiveness report, a levels by items summary, and a statistical summary report.

3. Input Data

Input data for ACIM is organized in the following three data sets:

Systems Factors

Format A - Options and Default Values
Format L - Site data

Item Data - Format I

One other data set can be used as an option, the Additional Item Data set, which further defines individual parts with respect to MSRT and repair cycle times, and also provides for user inserted site provisioning stocks at up to ten sites. Data elements for each of the three data sets along with an example of each record are provided on the following pages.

Format A - Options and Default Values. There is one record in this format. Figure 5.1 shows the Format A data elements along with a sample record. Data elements are defined as follows:

Format A - Option and Default Value Data Elements

COLS	DATA ELEMENT	UNITS
1	Format Identification (A)	
3-16	Run identification	
18-27	Run options	
28-31	Equipment MTTR	Days
32-35	Availability target	Fraction
37-43	Investment target	$000
	C-E Report Controls	
45-47	Units	
48-51	Availability	Fraction
52-56	Investment	
58-59	Part number field size	
	User MSRT	
62-65	Navy	Days
66-69	DLA	Days
70-73	Depot procurement leadtime	Days
74-76	Depot repair cycle	Days
77-80	Scrap rate	Fraction

Format A Record Example

```
A E2C                      .083.999     420
0000000001111111111222222222233333333334444
1234567890123456789012345678901234567890123
```

```
    1            17.517.5 360 83 .10
4444445555555555666666666677777777778
4567890123456789012345678901234567890
```

Figure 5.1. Format A Data Elements and Record Example.

Format Identification. An "A" is inserted in the first column to identify this data as format A.

Run Identification. Text entered in this field is printed at the top of all output reports to identify the particular run of the model.

Options. Entries in these fields control various features or operations of the model. Currently, the first four of the ten option fields are defined as follows:

a. MEC input type.

b. MEC use.

c. Default MSRT.

d. Levels format.

Equipment MTTR. Enter MTTR in days. This is the time required to accomplish the repair assuming all required repair parts are immediately available.

A_o Target. Enter the operational availability target as a fraction (including the decimal point). The model will build up stockages until this target or the investment target is reached. Enter .99 if reaching the investment target first is desired.

Investment Target. Enter the investment target, in thousands of dollars, in this field. Enter a large number (e.g., "9" in all columns) if reaching the A_o target first is desired.

Cost Effectiveness. These fields are used to control the production of the Cost-Effectiveness Report. As a unit

is added to stock, a line of data may appear on the Cost-Effectiveness Report if any one of the conditions based upon the following data occurs:

a. Delta Units. A line of data is produced for every nth unit added to stock, where n is specified in this field.

b. Delta A_o. A line of data is produced whenever the achieved A_o exceeds an integral multiple of this value.

c. Delta $. A line of data is produced whenever the achieved investment first exceeds an integral multiple of this value.

Part Number Field Size. In the Part Number/Nomenclature field of the Item Data Records, the left-hand side is used for Part Number and right-hand side is used for Nomenclature. The number of positions used for the Part Number is specified in this field.

Response Times. The average length of time, in days, required for a user of the equipment to obtain resupply from a higher supply source. One entry is for Navy COG items and one for DLA COG items. CNO current policy is to enter a value of 17.5 days for both items.

Depot PLT. A default value for depot procurement lead time (total time required to procure material from a manufacturer) is entered here, in days. This value is used whenever the PLT field in the Additional Item Data file is left blank.

Depot Repair Cycle. A default value for the depot repair cycle, in days, is entered in this field. This value is used whenever the depot repair cycle field in the Additional Item Data file is left blank.

Scrap Rate. A standard scrap rate is entered in this field as a fraction (e.g., 0.05). This is used as a default whenever the corresponding field in the Additional Item Data file is left blank.

<u>Format L - Site Data</u>. There is one record in the "L" format for each different kind of user or higher level maintenance/ supply activity in the support system for the equipment. Figure 5.2 shows the Format L data elements along with a sample record. In this study only one level is examined, and so only one Format L Record is entered.

Identification. An "L" is entered in column 1 to identify this format.

Site Name. Enter any text that identifies the site.

Indenture Level. Enter 1 for a single echelon case.

Echelon Code. Enter 0 for organization site.

Stockage Facility. Enter any mark if the site maintains inventory levels. For this study the carrier maintains inventory.

Repair Facility. Enter any mark if the site performs maintenance. For this study the carrier AIMD performs it.

Format L - Site Data Elements

COLS	DATA ELEMENT	UNITS
1	Format Identification (L)	
3-16	Site Name	
18	Indenture Level	
20	Echelon Code	
22	Stockage facility	
24	Repair facility	
26-29	Lead time	Days
31-34	Repair Cycle	Days
36-38	No. of locations	
40-42	No. of equipments	
44-45	Comparison policy	
47-48	ACIM Policy	
50-54	Availability target	Fraction
56-69	Operating factor	Fraction
61	Levels output format	

Format L Record Example

```
L CVN            1 0 X               1
0000000001111111111222222222233333333334
1234567890123456789012345678901234567890

 1  2  0
4444444445555555555666666666677777777778
1234567890123456789012345678901234567890
```

Figure 5.2. Format L Data Elements and Record Example.

Lead Time. The average length of time required, in days, for this site to obtain resupply from a higher supply source assuming that supplies are immediately available at the supply source. Enter 17.5 days.

Repair Cycle. Enter the average repair cycle, in days, for items that are normally repaired at this site.

Number of Locations. Enter the number of different users at this site (one for this study).

Comparison Policy. Not Applicable.

ACIM Policy. Code "O" for Optimization (ACIM starts all stocks at zero).

Operating Factor. Leave blank.

Levels Output Format. Leave blank.

Format I - Item Data. There is one record of this format in the Item Data file for each item in the equipment parts breakdown. The first record must always represent the equipment as a whole. Figure 5.3 shows the Format I data elements along with a sample record. The data elements are defined as follows:

Identification. Enter an "I" to identify this format.

Reference Number. The entry in this field is used to identify the item and its position in the parts breakdown of the equipment. Optional entry.

Indenture. The first record, representing the equipment as a whole, must have an Indenture Code of 1. All candidates

Format I - Item Data Elements

COLS	DATA ELEMENTS	UNITS
1	Format identification (I)	
2-11	Reference number	
12	Indenture	
14-42	Part number/nomenclature	
43-44	Cognizance	
45-50	Number per next higher assy	$/cents
60-64	SMR&R codes	
65-71	Best Replacement factor	per year
72-75	Minimum Replacement unit	
76	Military essentiality code	
77	Override code	
78	Override amount	

Format I Record Example

```
1       1072          HF POWER AMP
000000000111111111122222222223333333333
123456789012345678901234567890123456789
```

```
   1R  6    3494000  0G  8.4050    11
4444444444555555555566666666667777777777 8
0123456789012345678901234567890123456789 0
```

Figure 5.3. Format I Data Elements and Record Example.

after the first should be assigned a code of 2.

Part Number. Enter the NIIN/NICN or other part or stock number for item identification purposes. Part number field size is defined in Format A. The rest of the field entries are for Nomenclature.

Nomenclature. Enter textual data that identifies or describes the item.

Cognizance Code. Enter a code identifying the management cognizance of the item.

Number Per Next Higher Assembly. Enter the number of units of the item in the equipment.

Unit Cost. Enter the estimated unit procurement cost of the item in dollars and cents. There is an implied decimal point between columns 57 and 58 (cents occupy columns 58-59).

SM&R Codes. The Source, Maintenance and Recoverability codes are given. Entries for the maintenance codes are mandatory, others are optional.

Application Replacement Factor. Enter the actual anticipated number of times that the item will be replaced during one year of operation. This value represents an average over all items (of this type) in the system.

Minimum Replacement Unit (MRU). Enter a value for the MRU if different than 1. For this study 1 was used.

Military Essentiality Code (MEC). Enter a 1.

Override Code. The only override code used was Y, which was assigned to indenture level 1 equipment (total system). This code includes the item in all model processes but a zero stock level is assigned.

Override Quantity. Not applicable.

C. MODEL LIMITATIONS

The ACIM model is the most flexible model of the three inventory models discussed so far. ACIM's flexibility lies in its ability to solve either of the following problems for multi-echelon or single echelon supply systems:

1. Select a minimum cost collection of spares for a system so that the system will achieve a given availability target.

2. For a given budget select a collection of spares that will produce maximum availability for the system.

For this study ACIM was usually operated with a budget constraint. This was primarily due to the fact that the RIMAIR algorithm provided for control of the budget only. Therefore, the two models were compared on a equal budget basis. After running ACIM at a specific budget level the resulting inventory levels are manually input into the input data file for the TIGER simulation model.

ACIM, like the other two models, is a steady-state model. This means that the model operates on the assumption that all flows through the repair and requisition

pipelines have stabilized. The inventory system is assumed to be operating at a constant rate over a long period of time. This means that the model cannot be used to investigate surge demand periods.

This model does have a few computational approximations that should be noted. The first concerns ACIM's approximation of availability. ACIM assumes that no failures can occur after the first failure occurs. In actual aircraft systems, a single part failure will usually only degrade the system performance rather than cause the entire system to shut down. Parts usually continue to operate and continue to experience failures after one part fails. In addition, the process of minimizing MSRT does not yield the same stockage decisions as maximizing availability. For some systems the results may be similar, but for other systems there may be large differences.

Another peculiarity of the model is that it assumes that the yearly operating tempo input for a system represents operating tempo per "available year". For example, if an aircraft is scheduled for 1000 flight hours per year and 50% availability target is assigned, ACIM tacitly assumes it flies 500 hours per year.

When using the ACIM model to match a target budget (or availability), the iterative process only approximates the target goal. The ACIM algorithm will always exceed the target because it adds an item to the inventory until the

target is reached. Due to the discrete nature of the problem, the budget goal may be exceeded by an amount almost equal to the least expensive part; and that may be significant.

The ACIM model does present a significant increase in the workload required for data input. The exact topdown breakdown of parts, parts parameters, and maintenance facility information is required. Nevertheless, ACIM appears to be a useful tool and can be expanded to encompass many repair facilities at different levels, handling inventory problems of very complex systems.

VI. TEST RESULTS

A. INTRODUCTION

This chapter presents TIGER simulation results evaluating availability performance using inventory levels generated by the three models. The parts used for this evaluation were arranged in two systems which had identical part lists but which had different configurations.

The three inventory models were examined in three scenarios. The three topics to be covered are:

Fixed Budget. Achieved AVCAL availability is compared among the three inventory models, using a fixed budget constraint for each model.

Variable Budget. Availability is compared between the RIMAIR and ACIM models, while varying the budget over a range of values.

Variable MSRT. The ACIM model availability is analyzed with a variable MSRT parameter.

For each of the three tests it was assumed that spares decisions would be made for three identical aircraft systems. Availability would be computed for a period of ninety days, with each aircraft flying a total of 540 hours during the period. Each day was divided into the following four phases: a 3 hour flight phase, a 9 hour repair phase, a 3 hour flight phase, and a 9 hour repair phase. Aircraft

operated simultaneously during the flight phase, and were repaired simultaneously during the repair phase. This was considered an artificiality that was forced on the author by the nature of the TIGER simulation program (more realistic simulation would allow the aircraft to fly at different times during the day). For each test run, 25 iterations were done.

The system was composed of eight different part types with a total of fourteen individual parts. The size of the system and the number of aircraft used were limited by the TIGER program and by computer time limitations. Table III lists the part parameters according to part type.

TABLE III

Part Parameters

Part Type	Unit Cost	# Per A/C	BCM Rate	MTBF
1	34940	2	0.103	257
2	13670	2	0.180	352
3	10550	1	0.105	658
4	21930	1	0.247	667
5	37500	3	0.112	272
6	3520	1	0.238	699
7	38850	1	0.118	196
8	5060	3	0.258	1124

Note: BCM Rate is fraction of repairs that result in BCM action. MSRT in days.

Aircraft parts were arranged in two different configurations, corresponding to system configurations defined in OPNAVINST 5442.4H. The first configuration places the parts in series, corresponding to aircraft mission Code C, Full Fleet Defense. As defined in this mission, all WRA's are required to be in an operational (up) status for the aircraft to be capable of performing its mission. The second configuration corresponds to aircraft mission Code D, Expanded AAW Control. Here the requirements for WRA operational availability are more complex than the series layout. Basically there is some redundancy built into the second system, referred to as the parallel system, shown in Figure 6.1.

System Configuration (By Part Type) for Mission Code D

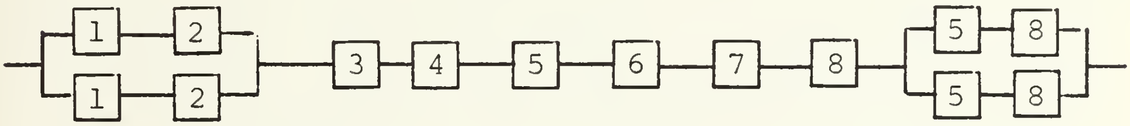


Figure 6.1. System Configuration (Mission Code D).

Aircraft parts were chosen from part candidates in the E-2C aircraft under the following criteria:

* Each part comprised an individual WRA.
* Each part was listed on the E-2C Mission Essential Subsystem Matrices (MESM).
* Most parts had low MTBFs.
* Each part was coded for removal at the O level and repair at the I level.

B. FIXED BUDGET ANALYSIS

The object of this portion of the study was to examine the effectiveness of the individual inventory models when all three models were constrained to the same budget level. The variable inputs to the TIGER program were the different AVCAL levels computed by each inventory model. Using TIGER generated aircraft availability, the capabilities and weaknesses of each model algorithm were examined. Both the series and parallel system configurations were used with each of the three inventory models.

A serious attempt was made to ensure that the three models were compared on an equal basis. This proved to be a difficult problem because of the differences among the models. The major difficulty was trying to equate the part parameters across the three models. The RIMAIR and ASO models are very similar so the parameters were matched for these two models first. Both models are subroutines of the TIGER program, and both use the same data input file. The ACIM model parameters were then matched to the ASO/RIMAIR parameters.

Since the ACIM model recommended using a benchmark value of 17.5 days for MSRT, this value was converted to hours (420) and used for both the Order and Shipping Time and the Repair Time in the RIMAIR/ASO models (SRTIM and REPTIM variables in TIGER). ACIM does not utilize these separate pipeline times, but only considers a single supply delay time (MSRT). A Mean Time to Repair equal to 0.083 days was input for ACIM, and the equivalent (2.0 hours) was input to TIGER. All other parameters were as listed in Table III.

The next step was to generate AVCAL stock levels for all three models at a fixed budget level. This was accomplished by first using the ASO algorithm to arrive at a benchmark budget level. Using the parameters discussed above, the total ASO inventory cost was $673,280. Next the RIMAIR model was run, varying the lambda value to arrive at a total inventory cost that was close to the ASO budget. The total RIMAIR inventory cost was $668,700 (within 1% of ASO budget). The ACIM target budget was set to a value equal to that of the ASO budget, and the ACIM model arrived at a total inventory cost of $675,300.

The resulting availability figures for both system configurations are summarized in Table IV. The effectiveness of both the RIMAIR and ACIM models appears to be better than that of the ASO model. ACIM seems to perform better in the

series system than RIMAIR, with both about equal in the parallel system.

TABLE IV

Fixed Budget Summary

		No. of Parts Stocked by Type							
Model	Budget	1	2	3	4	5	6	7	8
ASO	$673,280	5	5	1	1	7	1	3	3
RIMAIR	$668,700	4	5	2	3	5	3	4	4
ACIM	$675,300	5	5	2	2	6	3	3	3

Model Availability

	Series		Parallel	
Model	AVA	AVMUP	AVA	AVMUP
ASO	0.5643	0.5304	0.6963	0.6788
RIMAIR	0.6358	0.5950	0.8562	0.8348
ACIM	0.7611	0.7129	0.8312	0.8120

The poor performance of the ASO model in comparison to the other two models can be explained by examining the inventory decisions made by the ASO algorithm. First, the critical parts of the series system are found by examining the Critical Equipments list on the ASO model TIGER output. The results of this list are summarized along with a part budget breakdown in Table V. The two most obvious oversights are denoted by the starred rows. ASO spent only 2.09% of the

total budget on part types #3 and #6. Yet these two part types together accounted for 37.08% of the total unavailability of the system. The ASO model failed to observe that these two lower priced, high MTBF WRAs were availability bargains compared to the more expensive, low MTBF WRAs, such as part type #5.

TABLE V

Critical Equipments Analysis of the ASO Model

Part Type	Part Cost	# Stocked/ % Total Budget	Part Contribution To System Unavailability
3	$10550	1 / 1.57%	21.48% *
7	$38850	3 /17.31%	15.98%
6	$ 3520	1 / 0.52%	15.60% *
1	$34940	5 /25.95%	15.25%
5	$37500	7 /38.99%	13.52%
4	$21930	1 / 3.26%	11.39%
2	$13670	5 /10.15%	5.76%
8	$ 5060	3 / 2.25%	1.02%

Further analysis of the individual ASO stock levels shows that part types #3, #4, and #6 were stocked to a level of only one unit. These three parts qualified for a single spare each under the rotatable pool criteria, but none of the parts qualified for a spare under the attrition allowance portion of the AVCAL. This minimum stock level for these three parts was the major contributor to the poor performance of the ASO model. The ASO model failed to include unit cost

or cost effectiveness tradeoff analysis in computing stock levels, instead inventory levels were decided totally on the basis of MTBF and TAT.

For each case studied, the computed measure of availability, AVA, was several percentage points higher than AVMUP. As noted in Chapter II, AVMUP measures only the availability of the system during flight hours. AVMUP does not consider the operational status of the system during the repair (ondeck) phase. For the next two topics, AVMUP will be used as the primary measure of effectiveness.

C. VARIABLE BUDGET ANALYSIS

In the previous section each AVCAL model was studied at a single specified budget level. A more important question concerns the performance of these models over a range of budget levels. With an increase or decrease in budget level, the decision maker must adjust AVCAL levels accordingly. The ASO model was not included in this analysis since it does not lend itself easily to a variable budget analysis and because the ASO Manual does not provide any guidance for adjusting levels. It was also decided that the lack of any cost effectiveness measures in the ASO algorithm would cause the model to perform poorly at all budget levels.

With only the RIMAIR and ACIM models to compare, the test was arranged as follows. Part parameters remained as depicted in Table III, and mission times also remained the same.

Budget levels were varied from the benchmark budget used in VI.B above. Using the benchmark budget of $668,700 as 100%, test budgets were varied from a low of $521,810 (63%) to a high of $797,370 (119%). As before, the RIMAIR model was run first, varying the lambda value to arrive at an appropriate budget level. Using the resulting RIMAIR budget as a target budget, ACIM was then run. Inventory levels computed at each budget level are summarized in Table VI.

Using these inventory levels, model effectiveness was studied using both the series and parallel systems. Table VII summarizes system availability (AVMUP) for both models over the range of budgets. Budget percentages listed are those from the RIMAIR case. The average difference between the RIMAIR budget and the ACIM budget was 0.63%, with a maximum difference of 1.11%.

The results of this test were that both inventory models achieved similar operational availability. At lower level budgets the RIMAIR model did somewhat better than the ACIM model. Starting at about 80% of the benchmark budget level, ACIM performed equally well, and sometimes better. ACIM performed much better than RIMAIR at the 100% level for the series system. Considering the variability of the TIGER outputs, no model could be considered superior for all budget levels.

TABLE VI

AVCAL Stock Levels For Variable Budget

RIMAIR Model

Total Cost	% Benchmark	Stock Level By Part Type							
AVCAL	Budget	1	2	3	4	6	6	7	8
$423,210	63.29 %	3	3	2	2	3	2	2	3
$521,810	78.03 %	3	4	2	2	4	3	3	4
$578,680	86.54 %	4	4	2	3	4	3	3	4
$629,850	94.19 %	4	5	2	3	5	3	3	4
$708,700	105.98 %	5	5	2	3	5	3	4	5
$759,870	113.63 %	5	6	2	3	6	3	4	5
$797,370	119.24 %	5	6	2	3	7	3	4	5

ACIM Model

Total Cost	% Benchmark	Stock Level By Part Type							
AVCAL	Budget	1	2	3	4	5	6	7	8
$423,170	63.28 %	3	3	1	1	4	2	2	2
$524,890	78.49 %	4	4	2	1	5	2	2	3
$585,670	87.58%	4	4	2	2	5	2	3	3
$626,690	93.72 %	4	4	2	2	6	3	3	3
$714,150	106.80 %	5	5	2	2	6	3	4	3
$767,260	114.74 %	5	5	3	2	7	3	4	4
$802,200	119.96 %	6	5	3	2	7	3	4	4

D. VARIABLE MSRT ANALYSIS

For this portion of the study, the MSRT input to the ACIM model was varied in order to investigate the effect of a variable resupply and repair time on the effectiveness of a fixed budget inventory model. This analysis did not include the RIMAIR model because of the difficulties involved

TABLE VII

RIMAIR vs. ACIM Performance For Variable Budget

% Benchmark BUDGET	SERIES AVMUP RIMAIR	SERIES AVMUP ACIM	PARALLEL AVMUP RIMAIR	PARALLEL AVMUP ACIM
63.29 %	0.3657	0.2960	0.6334	0.5152
78.03 %	0.4519	0.4461	0.7678	0.5696
86.54 %	0.5391	0.5579	0.7591	0.7941
94.19 %	0.6089	0.6029	0.8378	0.7867
100.00 %	0.5950	0.7129	0.8348	0.8120
105.98 %	0.6821	0.7552	0.8678	0.8824
113.63 %	0.7122	0.7733	0.8343	0.8850
119.24 %	0.7964	0.7764	0.8813	0.8741

with keeping RIMAIR's budget constant while the supply times were being varied. Each time the supply time was changed in the RIMAIR model, a new lambda value had to be found to keep the inventory cost near the target budget. This proved extremely difficult because of the sensitivity of the lambda value to changes in supply time.

The methodology used for this test was as follows. The MSRT input parameter in ACIM was varied from 12 to 41 days, while maintaining a constant target budget of $544,000. Inventory levels were computed and then run on the TIGER program. TIGER parameters for repair and resupply times were matched to the corresponding MSRTs used in the ACIM program. System availability was analyzed for the series system only.

The ACIM Statistical Summary Report includes an achieved operational availability figure that theoretically could be achieved for a series system, given the inventory levels selected. These availability predictions, along with availabilities calculated from TIGER simulations, are compared in Table VIII. Several items should be noted regarding these results. ACIM projected availability is overly optimistic for low MSRT values, and then under-estimates availability for high MSRT values.

Nother noteworthy item was that the ACIM AVCAL stock levels varied slightly depending on the MSRT. When MSRT was increased, inventory levels for part types #4 and #7 decreased by one unit each, while part types #2 and #5 increased by 1 unit and 2 units respectively. The reason for this change is not clear. Once again it was observed that ACIM only approximates its target budget. As MSRT increased, ACIM overshot its target by a greater margin. Initially ACIM was within $4,000 of target, but this margin jumped to $32,000 (6% over target) at the maximum MSRT value of 41 days.

TABLE VIII

ACIM Performance For Variable MSRT

MSRT (DAYS)	ACIM Forecast Availability		TIGER AVMUP	AVCAL By Part Type 1	2	3	4	5	6	7	8
12	0.7922	OVERLY	0.6821	4	4	2	2	4	2	3	3
17.5	0.607	OPTIMISTIC	0.5359	4	4	2	1	5	2	3	3
24	0.365		0.3422	4	4	2	1	5	2	3	3
30	0.224	UNDER	0.2953	4	4	2	1	5	2	3	3
36	0.145	EST MATL AVAIL	0.3086	4	4	2	1	6	2	2	3
41	0.112		0.2535	4	5	2	1	6	2	2	3

Note: ACIM Target Budget = $544,000
12 Day MSRT Inventory Cost = $548,000
41 Day MSRT Inventory Cost = $576,000

VII. SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

A. SUMMARY AND CONCLUSIONS

Three areas of study were covered in testing inventory model effectiveness. The first area, fixed budget analysis, showed dramatic differences in model effectiveness. Budget allocation for the ASO model was much less efficient than either the RIMAIR or ACIM models. The ASO model, unlike the other two, made no attempt to determine optimum allocation of monetary resources. The ASO model also has no provisions to increase or decrease inventory levels according to budgetary constraints, except by manual additions or deletions to individual part stocks.

Examination of the Critical Equipments Summary of TIGER was useful in discovering inventory model weaknesses. This summary provides a list of parts that contribute to system downtime. Parts are listed according to the percentage of downtime that each part contributed. By matching these downtime percentages against part budget allocation percentages, inventory decisions can be evaluated.

The second area of study examined the RIMAIR and ACIM models in a variable budget analysis. Budget levels were varied from 63% to 120% of a benchmark budget level. Results of this test were inconclusive. Neither model showed complete

superiority at all budget levels. Similarities were noted in individual part stock levels of both models, resulting in similar availability statistics.

The third area of study concentrated on what effects varying MSRT had on ACIM effectiveness. Test results generally agreed with inventory theory. Availability decreased as MSRT was increased from 12 to 41 days. ACIM calculated availability differed from that derived in the TIGER simulation. Also noteworthy was the change in individual part stock levels as MSRT was increased. With a fixed target budget, ACIM had different part priorities depending on the length of MSRT.

This thesis investigated inventory model effectiveness as measured by aircraft system operational availability (AVMUP). One advantage to the use of AVMUP was that it allowed for simulated operation of aircraft systems in a degraded mode. That is, if one WRA failed the rest of the system WRAs continued to operate even if the system was in a down status. For complex aircraft systems the assumption of independent failures may not be a completely correct one, but it allows for simpler calculations.

Some of the advantages and disadvantages of each of the three inventory models are summarized below:

ASO Model.

Advantages.

1. Easy input data preparation.

2. Simple algorithm to determine stock levels (included as subroutine in TIGER).

Disadvantages.

1. Does not optimize budget allocation.

2. No provision for increasing or decreasing inventory levels to match budget constraints.

3. Separates pipeline demand into two parts. Low failure rate items may not qualify under either separate criteria.

RIMAIR Model.

Advantages.

1. Inventory levels can be varied to match budget constraints.

2. Provides protection for more portions of the inventory pipeline system than the ASO model.

3. Essentiality code allows for weighting of parts according to their relative criticality.

Disadvantages.

1. Lengthy process involved in changing lambda values to meet budget target when more than one model parameter is varied.

2. Model inherits weaknesses of ASO model because of identical model assumptions.

3. Without a comprehensive essentiality coding scheme, the optimization process only maximizes gross supply effectiveness. This process does not necessarily result in maximum availability.

ACIM Model.

Advantages.

1. Powerful model which has the capability to compute multi-echelon inventories for multi-indentured systems.

2. Stock levels can be determined for an availability target or a budget target.

3. Attempts to conform to recent CNO directives concerning inventory policies, especially maximizing availability.

Disadvantages.

1. Complicated algorithm is used to determine inventory stock levels.

2. Does not differentiate between repair and requisition pieplines.

3. Assumes that failure of one part results in shutdown of all other parts in system.

4. Optimization process only approximates maximization of availability by minimizing MSRT.

B. RECOMMENDATIONS

The TIGER model proved to be a capable evaluation tool, although it did have several limitations. Aircraft systems had to be simulated as if they were operated simultaneously. A more realistic simulation would allow for the overlapping of operating cycles. Sort routines in the TIGER TTE subroutine are not efficient and need to be improved. There are many TIGER calculations repeated every mission which are

not always applicable, but there is no easy method to eliminate these unnecessary steps. Input data preparation for TIGER can be tedious, especially for system configuration cards of complex systems.

Integration of the ASO and RIMAIR AVCAL inventory models into the TIGER model allowed for automatic AVCAL determination and simulation. Changes need to be included in the RIMAIR program to permit automatic adjustment of inventory levels to meet budget constraints. Manual adjustments of the lambda value to control budget levels was a slow process that prevented broader testing of the RIMAIR model.

Comparison of inventory model effectiveness must be done with some reservations. All three models assume steady-state inventory flows. The simulations were done in a manner to accommodate these assumptions. Different results may occur if surge demands or cyclic patterns are introduced into the simulation.

Another reservation involves the limited number of system configurations used in this study. None of the three inventory models take into account the configuration of the system. Additional research needs to be done to determine what effects alternate system configurations have on model inventory decisions. More study also needs to be done on the utilization of the item essentiality code parameters of the ACIM and RIMAIR models.

APPENDIX A

TIGER DATA CARD FORMATS

The following card formats were utilized in this study. Most card remain unchanged from the TIGER Manual, but some new cards are presented. Sample input data files are presented in appendices B and C. All data is entered in 80 column, card/card-image format. Data types are real, integer, and alphanumeric. All integer data fields must be right justified. Variable names listed are those that appear in this version of TIGER.

1. RIMAIR Parameter Card

New Card. Provides parameters for RIMAIR model algorithm. Budget parameter was not used. Essentiality code (ESS) was set to 1.0. Resupply delay time (RET) set to 0.

Columns	Format	Variable	Description
1-4	I4	NTOTA	Total no. of part types per A/C
5-8	F4.0	XFLAG	Used to select inventory policy: 0.0-Manual input of stock levels 1.0-ASO MANUAL policy 2.0-RIMAIR Policy
9-16	F8.0	BUDGET	Maximum budget constraint
17-31	F15.12	EL	Lagrange multiplier
32-36	F5.3	ESS	Essentiality code
37-42	F6.0	RET	Resupply delay time

---

2. Part Parameter Card.

New Card. One card is entered for each type equipment, I.

Columns	Format	Variable	Description
1-8	F8.2	COST(I)	Equipment unit cost.
9-16	F8.0	SRTIM(I)	Off-ship order & shipping time
17-21	I4	NPET(I)	No. of parts of type I per A/C
22-30	F8.4	BCM(I)	Fraction of parts BCM'ed

---

3. Flight Hours Data Card.

JTIME is the total time in a 90 day period that an A/C is expected to fly.

Columns	Format	Variable	Escription
1-8	I8	JTIME	Total flight times, summed over a 90 day period.

4. Timeline Iteration Card.

Columns	Format	Variable	Description
1-4	I4	JCC	No. of timeline iterations to be run for the data deck.
5-80	19A4	RUNID	Alphanumeric run identifier.

---

5. Statistical Parameter Card.

To run a predetermined # of missions, set NOPT & NMAX equal to the no. of missions, and PL = 1.0. A value of XK = 1.28 corresponds to 90% lower confidence limit.

Columns	Format	Variable	Description
1-4	I4	NMAX	Max no. of missions to be run (may not exceed 1000)
5-8	I4	NOPT	Optimal no. of missions to be run (may not exceed MAX)
9-12	F4.0	PL	Reliability spec. required
13-16	F4.0	XK	Std.Dev. for lower conf. limit
17-20	I4	ISEED	Random number seed
21-24	I4	NPH	No. of phase types (max of 6)

6. Phase Type and Duration Cards.

This card specifies the type of phase and duration of each phase. A phase is time period with both a repair policy and a system operation policy. Two phase types were used: type 1, flight phase; and type 2, ondeck repair phase. One card corresponds to a 24-hour period. Duration is in hours. System is presently configured for four phases per day.

Columns	Format	Variable	Description
1-2	F2.0	XXT(1)	Phase type no. of first phase
3-8	F6.0	XXT(2)	Duration of first phase
9-10	F2.0	XXT(3)	Phase type no. of second phase
11-16	F6.0	XXT(4)	Duration of second phase
17-18	F2.0	XXT(5)	Phase type no. of third phase
19-24	F6.0	XXT(6)	Duration of third phase
25-26	F2.0	XXT(7)	Phase type no. of fourth phase
27-32	F6.0	XXT(8)	Duration of fourth phase

7. Deployment Scenario Card.

This card determines the scenario under which a simulation can be run. The default values will allow TIGER to simulate a mission under the same conditions under which the inventory models will calculate planned inventory levels. So if the ASO model, for example, plans for a 1000 flight-hour quarter with pipeline times equal to ten days, TIGER will simulate a 90 day mission with these same parameters. By changing the default values on this card, inventory levels will be calculated under parameters entered on previous cards, but TIGER will simulate under conditions defined by this card. This permits investigation of inventory policy during periods of abnormally high tempo flight operations or lengthened pipeline times. NDAYS can be varied from 0-90 days. NWAR is a 2-state variable: 0 means previous inventory parameters will be used, 1 means wartime scenario and new parameters will be used. One of the new wartime parameters is BCMFAC, which increases the BCM rate for all parts. Another is REPFAC, which will increase the on-ship repair time for all parts. NTIME will be the higher system operational time for the war scenario, equal to the flight time expected for one A/C in 90 days.

Columns	Format	Variable	Description
1-4	I4	NDAYS	Length of scenario (0-90 days)
5-8	I4	NWAR	Sets wartime scenario: 0: original parameters (no change) 1: new values for SRTIM, BCM & NTIME will be computed
9-12	I4	NOAC	No. of A/C used in scenario
13-16	I6	NTIME	A/C flight hours in wartime
17-21	F5.1	BCMFAC	Fractional change in BCM rate
22-26	F5.1	REPFAC	Fractional change in on-ship repair times during war

8. Printout Option Card.

Columns	Format	Variable	Description
1-4	I4	KOPT	Printout option switch: 1: management summary printout 2: engineering summary printout 3: complete details printout (for debugging only) 4: disables input data printout 5: to specify printout using the KS variables (see below) 6: TIGER/MANNING complete details printout

If KOPT=5, select from the following output options as needed (otherwise leave the fields blank)

Columns	Format	Variable	Description
5-8	I4	KS(1)	= 1: Input data
9-12	I4	KS(2)	= 1: equip. downtime at time of mission failure
13-16	I4	KS(3)	=1: downtime at end of phase
17-20	I4	KS(4)	=1: abort messages
21-24	I4	KS(5)	=1: all events
25-28	I4	KS(6)	=1: ETIME matrix
29-32	I4	KS(7)	=1: not used
33-36	I4	KS(8)	=1: not used
37-40	I4	KS(9)	=1: not used
41-44	I4	KS(10)	=1: system & subsystem status
45-48	I4	KS(11)	=1: TIGER/MANNING debugging
49-52	I4	KS(12)	=1: status of all groups
53-56	I4	KS(13)	=1: downtime messages

9. Phase Repair Card.

This this study repair option 0 was used to simulate flight ops and repair option 2 simulated A/C on deck under repair.

Columns	Format	Variable	Description
1-4	I4	IFLAG(1)	Repair option for each phase type, up to 6: = 0 if onboard repair allowed in the phase = 1 if no on-board repair allowed in the phase = 2 on-board repair allowed but failure inhibited
5-8	I4	IFLAG(2)	
9-12	I4	IFLAG(3)	
13-16	I4	IFLAG(4)	
17-20	I4	IFLAG(5)	
21-24	I4	IFLAG(6)	

10. Repair Policy Card.

REPOL was set to 1.0. Normally it determines what fraction of repairs will be done on-ship. In this study this fraction was determined by BCM(I) instead. TAD2 specifies how long a system can operate in a down state before system failure. For this study mission allowable downtime = 0. XM and XT were set at their default values = 1.0.

Columns	Format	Variable	Description
1-4	F4.0	REPOL	Decimal fraction of repairs to be performed aboard ship
5-12	F8.2	TAD2	Mission Allowable Downtime
13-16	F4.0	XM	MTBF multiplier
17-20	F4.0	XT	MTTR multiplier

11. Equipment Type Cards.

These cards define the parameters for each type equipment. X is the time to replace a WRA from the A/C if a spare is on hand, arbitrarily set = 2.0 hours. V is used in this study to specify the onboard repair time at the AIMD level.

Columns	Format	Variable	Description
1-4	I4	I	Equipment type numbers, to be assigned sequentially, from 1 to a maximum of 200
5-20	4A4	DUM(J)	Equipment type description
21-28	F8.0	X	Mean time between failure
29-32	F4.0	Y	Mean time to repair/replace
33-36	F4.0	U	Duty cycle utilization
37-40	F4.0	V	AIMD part repair time
41-44	F4.0	W	Admin delay time (depot/ship)
45-58	I4	IDUM	Not used

---

12. *** Blank Card *** (Signals the end of equip. cards)

---

13. Equipment Cards.

These cards, one for each type equipment list individual parts by number, according to the equipment type. The first number is equipment type, the numbers following it on the same line are the individual parts for each type equipment.

Columns	Format	Variable	Description
1-4	I4	NTYPE	The type no. associated with the part numbers following it
5-8	I4	LOAD(1)	Part numbers, 19 per line max. Numbers begin at 1 and may not exceed 500. No gaps allowed in numbering parts.
9-12	I4	LOAD(2)	
.	.	.	
.	.	.	
.	.	.	
77-80	I4	LOAD(19)	

14. *** Blank Card *** (Signals the end of equipment cards)

---

15. Spares Model Card.

The only option used on this card was "999." (columns 21-24)

Column	Format	Variable	Description
21-24	F4.0	SX	Used to call Spares subroutine to determine allowance levels

---

16. ACIM Inventory Card.

This card will input ACIM allowance levels. If XFLAG=0.0 is selected on card #1, TIGER will simulate with this input. Any arbitrary inventory levels may be input on this line.

Column	Format	Variable	Description
1-2	I2	ISPARE	One allowance is entered for each equipment type, up to a max of 31.
3-4	I2	ISPARE	
.	.	.	
.	.	.	
.	.	.	
61-62	I2	ISPARE	

---

17. System Card.

Columns	Format	Variable	Description
1-4	A4	ID	Any alphanumeric e.g. SYST, to identify the specific system
5-8	I4	LL	Phase Type number (sequential) maximum value is 6
9-12	I4	NSS	No. of subsystems in the phase (varies only from 1 to 31)
13-16	I4	ISS	System identification number, usually last group number on the configuration matrix cards
17-24	F8.0	SSTIME	System allowable downtime. 100000 inhibits aborts.

18. Subsystem Cards.

One for each subsystem - up to 31. At least 1 subsystem is required.

Columns	Format	Variable	Description
1-4	A4	ID	Any alphanumeric, e.g., the literal SS1, SS2, ...SS31
5-8	I4	LL	Phase type number
13-16	I4	ISS	Subsystem identification no. This is a group # for a group defined on a Configuration Matrix Card. Each designated subsystem group must be a group that, upon its failure, causes the system to fail.
17-24	F8.0	SSTIME(2)	Subsystem allowable sustained downtime. To inhibit aborts use a value of 100000.

---

19. Configuration Matrix Card.

One card for each group, up to 300 cards.

Columns	Format	Variable	Description
1-4	I4	NRO	No. of members in the group defined on this card that are required to be operating and in an up status.
5-8	I4	IB(1)	The group no. assigned to the group of members defined on this card. It may vary from 501 to 1000 in any order.

Configuration cards cont'd.

Column	Format	Variable	Description
9-12	I4	IB(2)	The numbers of the equipment & groups which make up the group defined on this card. The max. no. of members in a group is unlimited; but if there are more than 7, a continuation card is required of the same format. The no. required and master group must be identical on all continuation cards.
13-16	I4	IB(3)	
17-20	I4	IB(4)	
21-24	I4	IB(5)	
25-28	I4	IB(6)	
29-32	I4	IB(7)	
33-36	I4	IB(8)	

---

20. *** Blank Card *** (Signals the end of phase configuration cards.)

NOTE: For each phase type, a set of the above System, Subsystem and Configuration Matrix Cards are entered, each set separated by a blank card.

---

21. Optional Output Card.

Columns	Format	Variable	Description
1-4	A4	SPRS	Place any alphanumeric, e.g., SPR, in this field if a table of spares usage is desired.
5-8	A4	APPL	Place any alphanumeric, e.g., APL, in this field if a summary table of equipment that caused mission failures (unreliability) and system down times (unavailability) is desired.
9-12	A4	GMMA	Not used
13-16	A4	DEMO	Not used

APPENDIX B

TIGER INPUT DATA FOR SERIES SYSTEM

This appendix contains the input data file representing the series system configuration. The TIGER program reads this file and proceeds with the simulation as defined in the data file. Part parameters for Appendix B are identical with those for the parallel system listed in Appendix C. Input data cards 17, 18, and 19 are the only data cards that are different for the series and parllel system data files.

```
   8 0.0 3000.0   0.000007519801.0      0.0
34940.    420.      2   0.103
13670.    420.      2   0.180
10550.    420.      1   0.105
21930.    420.      1   0.247
37500.    420.      3   0.112
3520.     420.      1   0.238
38850.    420.      1   0.118
5060.     420.      3   0.258
     540    1
   1 ACIM    675K 17C CODE83A    14AUG84
  25   25 1.01.281234    2
1.  3.0 2.  9.0 1.  3.0 2.    9.0
 90    0    3   540  1.0  1.0
  1
  0    2
 1.  0.00   1.  1.
   1HF POWER AMP        257.    2. 1.   420.
   2HF ANT COUPLER      352.    2. 1.   420.
   3HF RADIO CCNTRL     658.    2. 1.   420.
   4HF ANT CIU          667.    2. 1.   420.
   5ARC158 R/T          272.    2. 1.   420.
   6IFF R/T             699.    2. 1.   420.
   7IFF CODR/DECODR     196.    2. 1.   420.
   8ARC158 FLT CONV    1124.    2. 1.   420.

   1    1    2   15   16   29   30
   2    3    4   17   18   31   32
   3    5   19   33
   4    6   20   34
   5    7    8    9   21   22   23   35   36   37
   6   10   24   38
   7   11   25   39
   8   12   13   14   26   27   28   40   41   42

                            999.
 5  5  2  2  6  3  3  3
SYST    1    3  888  100000.
SS5     1       555  100000.
SS6     1       666  100000.
SS7     1       777  100000.
   7 511    1    2    3    4    5    6    7
   7 522    8    9   10   11   12   13   14
   2 555  511  522
   7 611   15   16   17   18   19   20   21
   7 622   22   23   24   25   26   27   28
   2 666  611  622
   7 711   29   30   31   32   33   34   35
   7 722   36   37   38   39   40   41   42
   2 777  711  722
   3 999  555  666  777
   1 888  999

SYST    2    3  888  100000.
SS5     2       555  100000.
SS6     2       666  100000.
SS7     2       777  100000.
   7 511    1    2    3    4    5    6    7
   7 522    8    9   10   11   12   13   14
   2 555  511  522
   7 611   15   16   17   18   19   20   21
   7 622   22   23   24   25   26   27   28
   2 666  611  622
   7 711   29   30   31   32   33   34   35
   7 722   36   37   38   39   40   41   42
   2 777  711  722
   3 999  555  666  777
```

1 888 999
SPRSAPPL

APPENDIX C

TIGER INPUT DATA FOR PARALLEL SYSTEM

This appendix contains the input data file representing the parallel system configuration.

```
   8 0.0 3000.0  0.000000001  1.0    0.0
34940.   420.      2   0.103
13670.   420.      2   0.180
10550.   420.      1   0.105
21930.   420.      1   0.247
37500.   420.      3   0.112
3520.    420.      1   0.238
38850.   420.      1   0.118
5060.    420.      3   0.258
     540    1
  1 ACIM   5675 FOR CODE83P 15AUG84
  25   25 1.01.281234   2
1.  3.0 2.  9.0 1. 3.0  2. 9.0
  90    0   3   540  1.0  1.0
  1
  0    2
 1.  0.00   1.  1.
   1HF POWER AMP       257.    2. 1.   420.
   2HF ANT COUPLER     352.    2. 1.   420.
   3HF RADIO CCNTRL    658.    2. 1.   420.
   4HF ANT CIU         667.    2. 1.   420.
   5ARC158 R/T         272.    2. 1.   420.
   6IFF R/T            699.    2. 1.   420.
   7IFF CODR/DECCDR    196.    2. 1.   420.
   8ARC158 FLT CONV   1124.    2. 1.   420.

   1    1    2   15   16   29   30
   2    3    4   17   18   31   32
   3    5   19   33
   4    6   20   34
   5    7    8    9   21   22   23   35   36   37
   6   10   24   38
   7   11   25   39
   8   12   13   14   26   27   28   40   41   42

                        999.
 5 5  2  2  6  3  3  3
SYST     1    4  888 100000.
SS5      1       555 100000.
SS6      1       666 100000.
SS7      1       777 100000.
SS9      1       999 100000.
   2 511    1    3
   2 512    2    4
   1 522  511  512
   6 533    5    6    7   10   11   12
   2 541    8   13
   2 542    9   14
   1 544  541  542
   3 555  522  533  544
   2 611   15   17
   2 612   16   18
   1 622  611  612
   6 633   19   20   21   24   25   26
   2 641   22   27
   2 642   23   28
   1 644  641  642
   3 666  622  633  644
   2 711   29   31
   2 712   30   32
   1 722  711  712
   6 733   33   34   35   38   39   40
   2 741   36   41
   2 742   37   42
   1 744  741  742
   3 777  722  733  744
   3 999  555  666  777
```

```
   1 888 999

SYST    2   4 888 100000.
SS5     2     555 100000.
SS6     2     666 100000.
SS7     2     777 100000.
SS9     2     999 100000.
   2 511   1   3
   2 512   2   4
   1 522 511 512
   6 533   5   6   7  10  11  12
   2 541   8  13
   2 542   9  14
   1 544 541 542
   3 555 522 533 544
   2 611  15  17
   2 612  16  18
   1 622 611 612
   6 633  19  20  21  24  25  26
   2 641  22  27
   2 642  23  28
   1 644 641 642
   3 666 622 633 644
   2 711  29  31
   2 712  30  32
   1 722 711 712
   6 733  33  34  35  38  39  40
   2 741  36  41
   2 742  37  42
   1 744 741 742
   3 777 722 733 744
   3 999 555 666 777
   1 888 999

SPRSAPPL
```

APPENDIX D

TIGER PROGRAM LISTING

This appendix contains a complete listing of the TIGER program as amended for this study. Input data is contained in a separate file as depicted in Appendix B. For an explanation of the changes made to this program see Chapter II. For a further explanation of the subroutines and options available to TIGER, see the TIGER Manual.

```
C
C
C  MAIN PROGRAM
C
      COMMON /ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFECP,T3,TIME,T3SUM
      COMMON/BETA/NRO(6,300),IB(6,300,8),NLINE(6)
      COMMON/EXTRA/ KS(20),ISW(31)
      COMMON /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON/NPH/ NSS(6),IFLAG(6),TITLE(6,31),SSTIME(6,31,2),ISS(6,31)
      COMMON /SEQ/INOABT(760),INMI(760),IAUP1(760),TT2(760),UP2(760)
     1,IAUP2(760)
      COMMON /TYP/EX(2,200),ISPARE(3,200),IUSED(3,200),IIUSED(3,200)
      COMMON /MAX/MAXNEQ,MAXTYP,MAXIB,MAXSTD
      COMMON/GAMMAA/XMTBA,VAR,RELGA(760),TIMA(760),XXT(1442),ITT,ISEED
      COMMON /TABORT/ XTABT(1000),RDT
      COMMON /TIGAP/ UP4,XNUM,BAPRIN,AVA,XPCAP,RUNID(19),TYCOON(500)
     +,COUNTB(500),XTCUM
      COMMON /DONE/DONE(3)
C     OREILLY ADD
C DELETING THEIR PRINTOUTS WITH CHERE
      COMMON /XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
      COMMON /KSPARE/JTIME,TOTSPR,COMB(9999),COMBA(9999),SER(360)
      COMMON /GEALG/NOSPRS
C SULLIVAN ADDS
      COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
      COMMON /ACSTAT/DTIME,UPLAST,JNDT
C SULLIVAN STOPS
      INTEGER TOTSPR,NOSPRS,COMB,COMBA,SER
C     OREILLY STOP
      DATA  BLNK/4H    /
C
      MAXRUN=1000
      MAXNPH=6
      MAXSTD=50
      MAXNEQ=500
      MAXTYP=200
      MAXIB=300
      MAXSS=31
C  SULLIVAN CHANGE
      MAXSEQ=760
C
C     CALL OVFLOW
C
C
C     OREILLY ADDS
C
```

```
C
      I=0
      NOSPRS=0
C   1 I=I+1
C     READ(5,9) COMB(I),COMBA(I),SER(I)
C     IF(COMB(I)-1) 10,3,3
C   3 IF(100-COMB(I)) 5,5,4
C   4 NOSPRS=NOSPRS+1
C   5 IF(100-COMBA(I)) 1,1,6
C   6 NOSPRS=NOSPRS+1
C     GO TO 1
C   9 FORMAT(I4,I4,I4)
C
C  10 READ (5,11) NTOTA,XFLAG,BUDGET
C  11 FORMAT(I4,F4.0,F8.0)
   10 READ (5,11) NTOTA,XFLAG,BUDGET,EL,ESS,RET
C     WRITE (6,11) NTOTA,XFLAG,BUDGET,EL,ESS,RET
   11 FORMAT (I4,F4.0,F8.0,F15.12,F5.3,F6.0)
C     READ (5,13) (COST(I),I=1,NTOTA)
C  13 FORMAT(F8.2)
C
      DO 14 I=1,NTOTA
        READ (5,13) COST(I),SRTIM(I),NPET(I),BCM(I)
   13   FORMAT (F8.2,F8.0,I4,F8.4)
   14 CONTINUE
C    SULLIVAN STOP
      READ(5,15) JTIME,TOTSPR
   15 FORMAT (I8,I4)
CM    WRITE (6,17) BUDGET
CM 17 FORMAT (//1X10HBUDGET IS ,F8.0)
C
C      OREILLY STOPS
      READ (5,19) JCC,(RUNID(I),I=1,19)
   19 FORMAT(I4,19A4)
C     WRITE (6,220) JCC
      DO 1230 JC=1,JCC
C  20 WRITE (6,30) (RUNID(I),I=1,19)
   30 FORMAT(1H1,30X,19A4//)
C     WRITE (6,40)
C     WRITE (6,50)
C     WRITE (6,55)
C     WRITE (6,56)
C     WRITE (6,57)
   40 FORMAT(1X,50H ++++++++++++++++++    TIGER  ++++++++++++++++++++ )
   50 FORMAT(/1X,50H++ NAVSEC 6112 LUETJEN+MANDEL+VAIL+ALLEY+BROWN ++ )
   55 FORMAT(/1X,50H++NPS IBM/360 VERSION LT. J. LEATHER THESES 9/80++)
   56 FORMAT(/1X,50H++AS AMENDED BY LCDR. P.J. O'REILLY THESES 12/81++)
   57 FORMAT(/1X,50H++AS AMENDED BY LCDR. M.D. SULLIVAN THESES  9/84++)
```

```
      BAPRIN=0.0
      DO 70 I=1,MAXNEQ
   60 COUNTB(I)=0.0
      TYCOON(I)=0.0
      KEQU(I)=0
   70 ETIME(I)=100000.
      NUM=0
      IFF=0
      IFR=0
      UP4=0.0
      T3=0.0
      T3SUM=0.0
      SUMX=0.0
      SUMX2=0.0
      DO 80 I=1,100
   80 TIMA(I)=0.
      DO 90 I=1,3
      DO 90 J=1,MAXTYP
C  SULLIVAN ADD
        REPTIM(J) = 100000.
        NOP(J) = 0
C
   90 IIUSED(I,J)=0
      DO 95 I = 1, MAXSS
        DO 95 J=1,20
          RFITIM(I,J) = 100000.
  95  CONTINUE
      TOTDT = 0.0
C SULLIVAN STOP
      DO 100 I=1,MAXSEQ
      TT2(I)=0.0
      UP2(I)=0.0
      IAUP1(I)=0
      IAUP2(I)=0
      REDAD1(I)=0.0
      INMI(I)=0
  100 INGABT(I)=0
      IAUP=0
      XTCUM=0
      IF (JC-1) 110,110,140
  110 READ (5,120) NMAX,NOPT,PL,XK,ISEED,NPH
  120 FORMAT (2I4,2F4.0,2I4)
  130 FORMAT (1X2I6,2XF4.2,2XF5.2,2XI6,2XI4)
  140 CONTINUE
CM160 WRITE (6,170) ISEED
  170 FORMAT (//1X15HRANDOM SEED IS ,I4)
      IF (NMAX-MAXRUN) 190,190,180
  180 NMAX=1000
```

```
      NOPT=1000
  190 DO 200 I=1,NMAX
  200 XTABT(I)=100000.
      WRITE (6,130) NMAX,NOPT,PL,XK,ISEED,NPH
      IF (MAXNPH-NPH) 1240,210,210
C  SULLIVAN CHANGE
  210 INUM=NCFT
C 220 FORMAT (/1X,5HJCC= ,4I10)
      READ (5,240)(XXT(I),I=1,8)
  240 FORMAT (4(F2.0,F6.0))
      READ (5,242) NDAYS,NWAR,NOAC,NTIME,BCMFAC,REPFAC
  242 FORMAT (I4,I4,I4,I6,F5.1,F5.1)
      WRITE (6,243) NDAYS,NOAC,NTIME
  243 FORMAT (2X,'NDAYS: ',I4,' NUMBER OF A/C ',I4,'NTIME ',I6)
      WRITE (6,244) BCMFAC,REPFAC
  244 FORMAT (2X,'BCMWAR ',F5.1,' REPAIR RATE CHANGE ',F5.1)
      NDMUL =(NDAYS* 8)-7
      NDSUB = NDAYS * 4
      IF (NDAYS-1) 250,250,245
C
C  THIS PORTION DEFINES THE PHASE TYPES AND PHASE INTERVALS.  THIS IS
C  SET JP FOR 4 DISTINCT PHASE INTERVALS FOR A 24 HOUR PERIOD.  THEN
C  THE 24 HOUR PERIOD IS REPLICATED FOR NDAYS (NORMALLY 90 DAYS).
C
 245  DO 250 I=9,NDMUL,8
         XXT(I) = XXT(1)
         XXT(I + 1) = XXT(2)
         XXT(I + 2) = XXT(3)
         XXT(I + 3) = XXT(4)
         XXT(I + 4) = XXT(5)
         XXT(I + 5) = XXT(6)
         XXT(I + 6) = XXT(7)
         XXT(I + 7) = XXT(8)
C        XXT(I + 8) = XXT(9)
C        XXT(I + 9) = XXT(10)
C        XXT(I + 10) = XXT(11)
C        XXT(I + 11) = XXT(12)
C        XXT(I + 12) = XXT(13)
C        XXT(I + 13) = XXT(14)
C        XXT(I + 14) = XXT(15)
C        XXT(I + 15) = XXT(16)
  250 CONTINUE
      XXT(I+8) = 0.
      XXT(I+9) = 0.
C  SULLIVAN STOP
C 260 WRITE (6,270)
  270 FORMAT (1H1,10X40HPHASE SEQUENCE  TYPE  DURATION  CUM TIME)
      IK=1
```

```
      IK2=2*IK
      IK3=IK2-1
      IXXT=XXT(IK3)
      TIMA(1)=XXT(2)
C     WRITE (6,280) IK,IXXT,XXT(IK2),TIMA(IK)
C 280 FORMAT (19XI4,2XI4,2XF8.2,2XF8.2)
C  SULLIVAN CHANGE
      DO 300 IK=2,NDSUB
C     DO 300 IK=2,100
      IK2=2*IK
      IK3=IK2-1
      IF (XXT(IK2)) 290,310,290
  290 TIMA(IK)=TIMA(IK-1)+XXT(IK2)
      IXXT=XXT(IK3)
C     WRITE (6,280) IK,IXXT,XXT(IK2),TIMA(IK)
  300 CONTINUE
C  SULIVAN STOP
  310 CONTINUE
      IF (JC-1) 320,320,330
  320 CONTINUE
C
      CALL PACK
C
C  SULLIVAN ACD
C
      IF (XFLAG - 1.0) 322,324,324
C
C  COMPUTE TOTAL COST OF ACIM INVENTORY PACKAGE
C
  322  COSTOT = 0.0
      DO 323 JA=1,NTYPE
             COSTOT = COST(JA)*FLOAT(ISPARE(1,JA)) + COSTOT
  323 CONTINUE
      WRITE (6,331) COSTOT
  331 FORMAT (2X,'TOTAL COST OF ACIM INVENTORY = ',F13.0)
C
C  NWAR IS A VARIABLE: NWAR=1 MEANS WAR-TIME SCENARIO, NWAR=0 PEACE.
C  IN THE WARTIME SCENARIO, NO OFF-SHIP SUPPLY IS ACCOMPLISHED, THE
C  BCM-RATE IS INCREASED BY BCMFAC, AND THE ON-SHIP REPAIR TIME IS
C  INCREASED BY REPFAC.
C
  324 IF (NWAR - 1) 330,325,330
  325 WRITE (6,326)
  326 FORMAT(2X,'WARTIME SCENARIO, NO AT-SEA RESUPPLY')
      DO 330 I = 1, NTOTA
         SRTIM(I) = 99999.
         BCM(I)   = BCMFAC * BCM(I)
         REPTIM(I) = REPFAC * REPTIM(I)
```

```
      WRITE (6,327) SRTIM(I),BCM(I),REPTIM(I)
327   FORMAT (2X,'SRT ',F9.2,' BCM ',F7.4,'REP ',F7.3)
C  SULLIVAN STOP
 330 CONTINUE
     JBB=1
     RELPY=1.0
     RELP=1.
     UP3=0.0
     TT3=0.0
     REDAD2=0.0
     DO 340 I=1,MAXSS
 340 ISW(I)=1
     ICRI=0
     DNT2=0.0
 350 STPHAS=0
     T1=0.0
C
C  SULLIVAN ADDS
C  DTIME IS ACCUMULATED TIME PER MISSION, FLOWN IN A DOWN STATUS.
C
     DTIME = 0.0
C
C  JNDT = 1 SIGNIFIES TO DTIME COMPUTATION THAT INFLIGHT FAILURES
C  OCCURRED, CTIME WAS COMPUTED UP TO AND INCLUDING THE END OF THE LAST
C  FLIGHT PHASE.  NO COMPUTATION IS NEEDED AT REPAIR PHASE START.
C
C
     JNDT = 0
     DO 355 JX = 1,NTOTA
       NOP(JX) = 0
       DO 355 JY = 1,20
          RFITIM(JX,JY) = 100000.
355  CONTINUE
C
C    RDT IS RUNNING DOWNTIME
C
     RDT=0.0
     IF (KS(8)) 380,380,360
 360 KAB=NUM+1
CHERE WRITE (6,370) KAB
 370 FORMAT(1X,16HSTART OF MISSION,I5,20H********************)
 380 KKK=0
 390 I=1
 400 LL=XXT(I)
     IF (LL) 450,450,410
 410 ENDPHA=STPHAS+XXT(I+1)
     I=I+2
C
```

```
      CALL RUN
C
      IX=NUM+1
      IF (XTABT(IX)) 420,420,440
CHERE ALSO CHANGE LABEL 420 BELOW
C 420 WRITE (6,430)
  430 FORMAT (1X44HTHE ABORT TIME IS ZERO,CHECK THE INPUT DATA.)
  420 GO TO 1240
  440 STPHAS=ENDPHA
      N=NSS(LL)+1
      GO TO 400
C
  450 NUM=NUM+1
C  SULLIVAN ADDS
      TOTDT = TOTDT + DTIME
C  SULLIVAN STOPS
      IF (IFFEOP) 460,460,480
  460 IFF=IFF+1
      IF (T3) 470,480,470
  470 CONTINUE
      T3SUM=T3SUM+T3
      T3=0.0
  480 XTCUM=XTCUM+XCUM
      UP4=UP4+ENDPHA-DNT2
      IF (XTABT(NUM)-100000.) 500,490,500
  490 X=ENDPHA
      GO TO 510
  500 X=XTABT(NUM)
  510 X2=X**2
      SUMX=SUMX+X
      SUMX2=SUMX2+X2
      IF (ISW(N)) 530,530,520
  520 IAUP=IAUP+1
  530 IF (NUM-INUM) 330,540,540
  540 INUM=INUM+50
C OREILLY ADD TO DELETE PRINTOUT EXCEPT FOR 1000TH MISSION.
      IF (NUM-NMAX) 570,550,570
  550 WRITE (6,560) NUM
  560 FORMAT (/1X16HA GRAND TOTAL OF,I6,24H MISSIONS HAVE BEEN RUN.)
  570 XNUM=NUM
  580 XPCAP=XTCUM/XNUM
C OREILLY ADD TO DELETE PRINTOUT EXCEPT FOR 1000TH MISSION.
      IF (NUM-NMAX) 610,590,610
  590 WRITE (6,600) XPCAP
  600 FORMAT (1X24HTHE RELIABILITY IS      ,F8.4)
  610 XPLCL=XPCAP-XK*SQRT(XPCAP*(1.-XPCAP)/XNUM)
      IF (XPLCL) 620,630,630
  620 XPLCL=0.0
```

```
C ORELLLY ADD TO DELETE PRINTOUT EXCEPT FOR 1000TH MISSION.
C    ADDRESS 631,661 ADDED FOR THIS PRINT DELETION ONLY
C    ADDRESS 630 ALSO MOVED FOR THIS DELETION
  630 IF (NUM-NMAX) 661,631,661
  631 WRITE (6,640) XPLCL
C 630 WRITE (6,640) XPLCL
  640 FORMAT (1X24HTHE LOWER CONF LIMIT IS ,F8.4)
      WRITE (6,650) PL
  650 FORMAT (1X24HTHE SPEC REQUIREMENT IS ,F8.4)
      WRITE (6,660) RED2
  660 FORMAT (1X17HTHE READINESS IS ,7XF8.4)
  661 AVA=UP4/TT3
C OREILLY ADD TO DELETE PRINTOUT EXCEPT FOR 1000TH MISSION.
C ADDRESSES 669 AND 671 ADDED FOR THIS PRINT DELETION ONLY
      IF (NUM - NMAX) 671,669,671
  669 WRITE (6,670) AVA
  670 FORMAT (1X28HTHE AVERAGE AVAILABILITY IS ,F8.4)
  671 XIAUP=IAUP
      AVAINS=XIAUP/XNUM
C OREILLY ADD TO DELETE PRINTOUT EXCEPT FOR 1000TH MISSION.
C ADDRESSES 679 AND 681 ADDED FOR THIS PRINT DELETION ONLY
      IF (NUM-NMAX) 681,679,681
  679 WRITE (6,680) AVAINS
  680 FORMAT (1X28HTHE INSTANT AVAILABILITY IS ,F8.4)
  681 XDWN=XNUM-XTCUM
      IF (XDWN) 690,690,700
  690 XMTBA=2.0*SUMX
      XLCLA=0.434*SUMX
      VAR=(0.5*SUMX)**2
      GO TO 710
  700 XMTBA=SUMX/XDWN
      VAR=(SUMX2/XNUM)-(SUMX/XNUM)**2
      CORR=(SUMX*(1/XDWN-1/XNUM))**2
      VAR=VAR+CORR
      XLCLA=XMTBA-(1.28*SQRT(VAR))
C OREILLY ADD TO DELETE PRINTOUT EXCEPT FOR 1000TH MISSION.
C ADDRESSES 741 ADDED FOR THIS PRINT DELETION ONLY
      IF (NUM-NMAX) 741,710,741
  710 WRITE (6,720) XMTBA
  720 FORMAT (/1X41HTHE MEAN TIME BETWEEN MISSION FAILURES IS,F20.1)
      WRITE (6,730) XLCLA
  730 FORMAT (1X21HTHE LCL,90, MTBMF IS ,F20.1)
      WRITE (6,740) VAR
  740 FORMAT (1X27HTHE MTBMF VARIANCE IS       ,F20.1)
  741 XIFF=IFF
      XIFR=IFR
      IF (IFF) 760,750,760
  750 XMUT=2.0*UP4
```

```
      XMDT=0.0
      GO TO 790
  760 XMUT=UP4/XIFF
      IF (IFR) 780,770,780
  770 XMDT=(TT3-UP4-T3SUM)/XIFF
      GO TO 790
  780 XMDT=(TT3-UP4-T3SUM)/XIFR
C OREILLY ADD TO DELETE PRINTOUT EXCEPT FOR 1000TH MISSION.
      IF (NUM - NMAX) 830,790,830
  790 WRITE (6,810) XMUT
C
C  SULLIVAN ADDS
C
      AVGDT  = TOTDT/FLOAT(NOPT)
      AVMUP  = 1 - (AVGDT/FLOAT(NTIME))
      WRITE (6,795) AVGDT
      WRITE (6,796) AVMUP
  795 FORMAT (2X,'AVERAGE HOURS FLOWN IN DOWN STATUS',F9.1)
  796 FORMAT (1X,'PROBABILITY OF ALL A/C IN AN UP STATUS ',F6.4)
C
C  SULLIVAN STOPS
C
  800 WRITE (6,820) XMDT
  810 FORMAT (/1X18HTHE SYSTEM MUT IS ,F20.1)
  820 FORMAT (1X18HTHE SYSTEM MDT IS ,F20.3)
  830 IF (XPCAP-PL) 840,840,920
  840 IF (NOPT-NUM) 870,870,850
  850 WRITE (6,860)
  860 FORMAT (1X14HANOTHER SET OF,3H 50,20HMISSIONS WILL BE RUN,43H TO O
     1BTAIN REQUIRED STATISTICAL CONFIDENCE.)
      GO TO 330
  870 WRITE (6,880)
  880 FORMAT (1X52HSIMULATION COMPLETE-OPTIMUM NUMBER MISSIONS WERE RUN)
      IF(PL.EQ.1.) GO TO 910
  890 WRITE (6,900)
  900 FORMAT (1X33HWEAPON SYSTEM FAILS REQUIREMENTS.)
  910 GO TO 1010
  920 IF (NMAX-NUM) 930,930,960
  930 WRITE (6,940)
  940 FORMAT (1X52HSIM COMPLETE-PREDEFINED MAX NUMBER MISSIONS WERE RUN)
  950 IF (XPLCL-PL) 890,990,990
  960 IF (XPLCL-PL) 850,970,970
  970 WRITE (6,980)
  980 FORMAT (2X22HSIMULATION COMPLETE - )
      IF(PL.EQ.1.) GO TO 1010
  990 WRITE (6,1000)
 1000 FORMAT (1X33HWEAPON SYSTEM MEETS REQUIREMENTS.)
 1010 CONTINUE
```

```
      IF (JC-1) 1020,1020,1040
 1020 READ (5,1030) SPRS,APPL,GMMA,DMNO
 1030 FORMAT (4A4)
 1040 IF (SPRS.EQ.BLNK) GO TO 1190
 1050 IDIFF=0
      TAFM=0.0
      TACMMH=0.0
      WRITE (6,1060)
 1060 FORMAT (1H1,4X53HEQUIP FAILURES AND CORRECTIVE MAINTENANCE(CM) SUM
     1MARY/8X71HEQUIP. NO.  TYPE NO.  TOTAL EQUIP.  AVG. NO. FAILURES  A
     2VG. CM MANHOURS/32X8HFAILURES,7X11HPER MISSION,5X11HPER MISSION/)
      DO 1090 I=1,NEQ
      IF(XMTTR(IEQU(I)).EQ.9999.) GO TO 1090
      IF (KEQU(I)) 1090,1090,1070
 1070 AFM=KEQU(I)/XNUM
      IEQ=IABS(IEQU(I))
      ACMMH=AFM*ABS(XMTTR(IEQ))
      WRITE(6,1080) I,IEQ,KEQU(I),AFM,ACMMH
 1080 FORMAT (10XI4,6XI4,6XI10,6XF10.3,6XF10.3)
      IDIFF=IDIFF+KEQU(I)
      TAFM=TAFM+AFM
      TACMMH=TACMMH+ACMMH
 1090 CONTINUE
      WRITE (6,1100) IDIFF,TAFM,TACMMH
 1100 FORMAT (31X10H----------,6X10H----------,6X10H----------/31XI10,6X
     1F10.3,6XF10.3)
 1110 CONTINUE
C     WRITE (6,1120)
C1120 FORMAT (1H1,3X41HAVERAGE NUMBER OF SPARES USED PER MISSION)
C     WRITE (6,1130)
C1130 FORMAT (/4X6HSPARES,7X4HSHIP,18X6HTENDER,16X4HBASE)
      WRITE (6,1140)
 1140 FORMAT (8X4HTYPE,4X3(5HSTOCK,3X4HUSED,10X))
      DO 1170 J=1,NTYPE
      ALDONE=0.0
      DO 1150 I=1,3
      DONE(I)=IIUSED(I,J)/XNUM
      ALDONE=ALDONE+DONE(I)
 1150 CONTINUE
      IF(ALDONE) 1155,1170,1155
 1155 WRITE(6,1160)J,(ISPARE(I,J),DONE(I),I=1,3)
 1160 FORMAT (8XI4,4X3(I5,F7.2,10X))
 1170 CONTINUE
 1180 CONTINUE
 1190 IF (APPL.EQ.BLNK) GO TO 1210
 1200 BAPRIN=-1.0
C
      CALL APPLE
```

```
C
 1210 CONTINUE
 1220 CONTINUE
 1230 CONTINLE
 1240 STOP
      END
C
C
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
C
C
      SUBROUTINE RUN
      COMMON /MAX/MAXNEQ,MAXTYP,MAXIB,MAXSTD
      COMMON /ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFECP,T3,TIME,T3SUM
      COMMON/BETA/NRO(6,300),IB(6,300,8),NLINE(6)
      COMMON/EXTRA/ KS(20),ISW(31)
      COMMON /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON/NPH/ NSS(6),IFLAG(6),TITLE(6,31),SSTIME(6,31,2),ISS(6,31)
      COMMON /SEQ/INOABT(760),INMI(760),IAUP1(760),TT2(760),UP2(760)
     1,IAUP2(760)
      COMMON /TYP/EX(2,200),ISPARE(3,200),IUSED(3,200),IIUSED(3,200)
      COMMON/GAMMAA/XMTBA,VAR,RELGA(760),TIMA(760),XXT(1442),ITT,ISEED
      COMMON /TABORT/ XTABT(1000),RDT
      COMMON /DELTA/KKK2
      COMMON /XXX/XXX
      COMMON/VDC/VDC(50,6),IUI(200),VMTTR(200,6),TAD2
      COMMON/STAN/ISTB(60,10,6)
      COMMON/RUNAP/ITEMP2,DELT,ISSA(31),ISSC
      COMMON/XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
C   SULLIVAN ADD
      COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
      COMMON /ACSTAT/DTIME,UPLAST,JNDT
      NDT = 0
C   SULLIVAN STOP
C
      TDEOP=0.0
      TP=STPHAS
C     WRITE (6,1)
C 1   FORMAT (2X,'RUN  ')
      KAA=NUM+1
      XKAA=KAA
      NX=NSS(LL)
      N=NX+1
      ITEMP=0
      ITEMP2=0
      IF (KKK) 40,10,40
```

```
   10 DO 20 I=1,3
      DO 20 J=1,NTYPE
      IUSED(I,J)=0
   20 CONTINUE
      DO 30 I=1,NEQ
   30 ETIME(I)=100000.
   40 CONTINUE
C
   50 DO 120 ILB=1,NEQ
      KEQ=ILB
      IF(ETIME(KEQ)+100001.001)55,120,55
   55 IF(ETIME(KEQ)+99999.)60,60,120
   60 IF (IFLAG(LL)) 120,70,120
C
   70 ETIME(KEQ)=STPHAS
      IABC=IABS(IEQU(KEQ))
      IF (XMTTR(IABC)) 80,80,100
   80 XXX=VMTTR(IABC,LL)
      IF (XXX-9999.) 120,90,120
   90 ETIME(KEQ)=-99999.
      GO TO 120
  100 XXX=XMTTR(IABC)
C
  110 CALL TTE
C
  120 CONTINUE
C
      DO 140 ILB=1,NEQ
      KEQ=ILB
      IEQU(KEQ)=IABS(IEQU(KEQ))
      IF (ETIME(KEQ)-100000.) 130,140,130
  130 IEQU(KEQ)=-IABS(IEQU(KEQ))
  140 CONTINUE
  150 CONTINUE
C
      KKK2=KKK
      K=NLINE(LL)
      DO 250 I=1,K
      DO 250 J=2,8
      KEQ=IABS(IB(LL,I,J))
      IF (KEQ-MAXNEQ) 151,151,250
  151 IF(KEQ)250,250,155
  155 IF(ETIME(KEQ)+100001.001)160,250,160
  160 IEQU(KEQ)=IABS(IEQU(KEQ))
      IABC=IEQU(KEQ)
      IF (XMTTR(IABC)) 170,170,180
  170 IF (VMTTR(IABC,LL)-9999.) 180,190,180
  180 CONTINUE
```

```
C
C  SULLIVAN CHANGE
C
C       IF (IFLAG(LL)-1) 210,190,210
C  190  IF (ETIME(KEQ)) 200,210,210
C  200  ETIME(KEQ)=ETIME(KEQ)-(ENDPHA-STPHAS)
C  210  IF (ETIME(KEQ)-100000.) 220,240,220
C  220  IF (ABS(ETIME(KEQ))-STPHAS) 240,230,230
C  230  IF (STPHAS) 250,240,250
C  240  ETIME(KEQ)=-STPHAS
C       IABC=IABS(IEQU(KEQ))
C       XXX=XMTBF(IABC)
C
   190  IF (ETIME(KEQ) - 100000.) 225,195,225
   195  ETIME(KEQ) = 0.0
        IABC = IABS(IEQU(KEQ))
        XXX = XMTBF(IABC)
        GO TO 240
   225  IF (IFLAG(LL) - 1) 250,250,230
   230  IF (ETIME(KEQ)) 250,250,235
   235  ETIME(KEQ) = ETIME(KEQ) + (ENDPHA - STPHAS)
        GO TO 250
   240  CONTINUE
C
C  SULLIVAN STOP
C
        CALL TTE
C
   250  CONTINUE
        KKK2=1
C
        DO 330 ILB=1,NEQ
        KEQ=ILB
        IF(ETIME(KEQ)+100001.001)255,330,255
   255  IF(IEQU(KEQ))260,260,330
   260  IEQU(KEQ)=IABS(IEQU(KEQ))
        IABC=IEQU(KEQ)
        IF (XMTTR(IABC)) 270,270,280
   270  IF (VMTTR(IABC,LL)-9999.) 280,290,280
   280  CONTINUE
        IF (IFLAG(LL)-1) 310,290,310
   290  IF (ETIME(KEQ)) 300,320,320
   300  ETIME(KEQ)=ETIME(KEQ)-(ENDPHA-STPHAS)
        GO TO 330
C
   310  IF(ETIME(KEQ)) 331,320,320
   320  ETIME(KEQ)=100000.
        IEQU(KEQ)=IABS(IEQU(KEQ))
```

```
      GO TO 330
  331 IEQU(KEQ)=-IABS(IEQU(KEQ))
  330 CONTINUE
C
      CALL STATUS
      CALL STNDBY
C
      CALL STATUS
C
C  SULLIVAN ADDS
C  THIS SECTION WILL CALCULATE DTIME IF AIRCRAFT STARTS FLIGHT PHASE
C  IN A DOWN STATUS.
C
      IF (ISW(N).GT.0) GO TO 338
      IF (IFLAG(LL) .GE.1) GO TO 336
      DTIME = DTIME + (ENDPHA - STPHAS)
      JNDT = 1
      GO TO 337
 336  IF (JNDT.EQ.1) GO TO 338
      DTIME = DTIME + (STPHAS - UPLAST)
      UPLAST = ENDPHA
      JNDT = 0
 337  NDT = 1
C     WRITE (6,478) DTIME, TIME
C
C  SULLIVAN STOPS
C
  338 IF (ISW(N)) 350,350,340
  340 IAUP1(JBB)=IAUP1(JBB)+1
  350 XIAUPI=IAUP1(JBB)
      XAVI=XIAUPI/XKAA
C
      TIME=STPHAS
      DNT1=0.0
      DO 360 KSS=1,N
  360 SSTIME(LL,KSS,1)=0.0
C
  370 TP=TIME
C
      CALL STNDBY
C
C  SULLIVAN ADDS
C
      IF (ISW(N) .GT.0) UPLAST = TIME
C
C  SULLIVAN STOPS
C
  380 IF (KS(6)) 390,440,390
```

```
  390 WRITE (6,430) TP
      DO 410 J=1,NEQ
      IF (ETIME(J)-100000.) 400,410,400
  400 IEQ=IABS(IEQU(J))
      WRITE(6,420) J,IEQ,ETIME(J)
  410 CONTINUE
  420 FORMAT (1XI5,1XI5,5XF22.4)
  430 FORMAT (1XF12.4)
C
  440 CALL EVENT
C
C     WRITE (6,445) ETIME(888)
C 445 FORMAT (2X,'ETIME (888) ',F6.1)
      TIME=ABS(ETIME(KEQ))
C
      IF (KS(5)) 450,470,450
  450 WRITE (6,460) KEQ,ETIME(KEQ),KAA
  460 FORMAT (10X5HEQUIP,I5,F12.4,5X7HMISSION,I10)
  470 DELT=TIME-TP
C
      CALL STATUS
C
C  SULLIVAN ADDS
C  THIS SECTION CALCULATES HOURS FLOWN IN A DOWN STATUS, WHEN AIRCRAFT
C  GOES TO A DOWN STATUS WHILE AIRBORNE.
C
C     IF (ISW(N)) 473,473,475
C473
C474  FORMAT (2X,'ISW(N) LESS THAN 1')
 475  IF (ISW(N).GT.0) GO TO 480
      IF (IFLAG(LL).GE.1) GO TO 480
      IF (NDT.EQ.1) GO TO 480
C     WRITE (6,992) ETIME(KEQ)
C992  FORMAT (2X,'ETIME ',F8.2)
C     IF (ETIME(KEQ).GE.ENDPHA) GO TO 480
C     WRITE (6,993) UPLAST
C993  FORMAT (2X,'UPLAST ',F8.2)
      DTIME = DTIME + (ENDPHA- UPLAST)
      NDT = 1
      JNDT = 1
C     WRITE (6,478) DTIME,TIME
C478  FORMAT (1X,'DOWNTIME: ',F8.2,' TIME: ',F9.1)
C
C  SULLIVAN STOPS
C
  480 DO 510 KSS=1,NX
      IF (ISW(KSS)) 490,490,500
  490 SSTIME(LL,KSS,1)=SSTIME(LL,KSS,1)+DELT
```

```
      GO TO 510
  500 SSTIME(LL,KSS,1)=0.0
  510 CONTINUE
      IF (ISW(N)) 520,520,530
  520 SSTIME(LL,N,1)=SSTIME(LL,N,1)+DELT
      T3=T3+DELT
      IF (TIME-ENDPHA) 522,522,521
  521 T3=T3+ENDPHA-TP-DELT
  522 RDT=RDT+DELT
      GO TO 550
  530 T3=0.0
      RDT=0.0
      IF (SSTIME(LL,N,1)) 1140,550,540
  540 T1=SSTIME(LL,N,1)
      SSTIME(LL,N,1)=0.0
  550 CONTINUE
C
      IF (SSTIME(LL,N,1)) 570,560,570
  560 IF (T1) 620,620,580
  570 IF (T1) 620,610,620
  580 IFF=IFF+1
  590 IFR=IFR+1
  600 T1=0.0
      GO TO 620
  610 T1=SSTIME(LL,N,1)
  620 CONTINUE
C
      IF(ICRI) 640,640,660
C
  640 ISSC=1
      ISSA(1)=N
      IF(RDT-TAD2)645,645,930
  645 ICRI=0
      IF(SSTIME(LL,N,1)-SSTIME(LL,N,2)) 650,650,960
  650 ICRI=0
      ISSC=0
      DO 655 KSS=1,NX
      IF(SSTIME(LL,KSS,1)-SSTIME(LL,KSS,2))655,655,652
  652 ISSC=ISSC+1
      ISSA(ISSC)=KSS
  655 CONTINUE
      IF(ISSC)660,660,962
  660 CONTINUE
      IF (TIME-ENDPHA) 670,670,1140
  670 IF (ISW(N)) 680,680,730
C
  680 CALL APPLE
C
```

```
  730 IF (ETIME(KEQ)) 810,810,740
  740 IABC=IABS(IEQU(KEQ))
      IF (IFLAG(LL)-1) 750,760,750
  750 CALL LRND(ISEED,RN,1,16807,0)
      IF (RN-REPOL) 770,770,800
  760 ETIME(KEQ)=-99999.
      GO TO 830
  770 IF (XMTTR(IABC)) 780,780,790
  780 XXX=VMTTR(IABC,LL)
      IF (XXX-9999.) 820,760,820
  790 XXX=XMTTR(IABC)
      GO TO 820
  800 ETIME(KEQ)=-100001.001
      GO TO 830
  810 IABC=IABS(IEQU(KEQ))
      XXX=XMTBF(IABC)
  820 IF(IEQU(KEQ))811,821,821
  811 IEQU(KEQ)=IABS(IEQU(KEQ))
C ?
      ETIME(KEQ)=100000.
      GO TO 830
C
  821 CALL TTE
C
  830 IF (ETIME(KEQ)) 840,1150,870
C
  840 KEQU(KEQ)=KEQU(KEQ)+1
      IF (ISW(N)) 850,850,370
  850 DNT1=DNT1+DELT
      IF (ICRI) 860,370,860
  860 REDAD1(JBB)=REDAD1(JBB)+DELT
      GO TO 370
C
  870 CONTINUE
      IF (ISW(N)) 880,880,370
  880 DNT1=DNT1+DELT
      IF (ICRI) 890,900,890
  890 REDAD1(JBB)=REDAD1(JBB)+DELT
  900 TDOWN=TIME-SSTIME(LL,N,1)
      TTEMP=SSTIME(LL,N,1)
      IF (KS(13)) 370,370,910
CHERE ALSO CHANGE LABEL 910
C 910 WRITE (6,920) LL,TDOWN,TTEMP,KAA
  920 FORMAT (13H DURING PHASE,I6,20HSYSTEM WENT DOWN AT ,F14.4,13H DOWN
     1TIME IS ,F14.4,3X7HMISSION,I6)
  910 GO TO 370
C
  930 ICRI=5
```

```
      TABORT=TIME-(RDT-TAD2)
      IF(TABORT-ENDPHA)940,645,645
  940 IF (XTAET(KAA)-100000.) 660,950,660
  950 ITEMP=1
      ITEMP2=1
CHERE WRITE(6,1010)LL,JBB,KAA,TABORT,TITLE(LL,N),TAD2
      GO TO 1020
  960 ICRI=4
      GO TO 964
  962 ICRI=2
  964 TABORT=TIME-(SSTIME(LL,ISSA(1),1)-SSTIME(LL,ISSA(1),2))
  970 IF (TABCRT-ENDPHA) 990,980,980
  980 IF(ICRI-2)650,985,650
  985 ICRI=0
      GO TO 660
C ?
  990 IF (XTABT(KAA)-100000.) 660,1000,660
 1000 ITEMP=1
      ITEMP2=1
CHERE DO 1005 I=1,ISSC
C1005 WRITE(6,1009)LL,JBB,KAA,TABORT,TITLE(LL,ISSA(I))
C    1,SSTIME(LL,ISSA(I),2)
 1009 FORMAT(1X9HIN PHASE ,I2,1X3HSEQ,I3,4X7HMISSION,I6,4X15HABORTED AT
     1TIME,F10.4,10H  BECAUSE ,A4,35H  EXCEEDED PHASE ALLOWABLE DOWNTIME
     2,2XF10.3,5H HRS.)
 1010 FORMAT (1X9HIN PHASE ,I2,1X3HSEQ,I3,4X7HMISSION,I6,4X15HABORTED AT
     1 TIME,F10.4,10H  BECAUSE ,A4,37H  EXCEEDED MISSION ALLOWABLE DOWNT
     2IME,2XF10.3,5H HRS.)
 1020 XTABT(KAA)=TABORT
      IF (TABORT) 1590,1590,1040
 1040 DO 1110 I=1,NEQ
      IF (ETIME(I)) 1050,1110,1110
 1050 IF (IEQU(I)) 1080,1110,1080
 1080 IF (KS(2)) 1110,1110,1110
CHERE PREVIOUS LINE WAS 1090,1110,1090
C1090 WRITE (6,1100) I,ETIME(I)
 1100 FORMAT (17X9HEQUIPMENT,I5,24H DOWN IT WILL COME UP AT,F16.4)
 1110 CONTINUE
C
 1120 CALL APPLE
C
      ITEMP2=0
 1130 GO TO 660
C
 1140 CONTINUE
      IFFEOP=ISW(N)
      IF (ISW(N)) 1160,1160,1270
 1150 CONTINUE
```

```
 1160 TDEOP=ENDPHA-TP
 1170 CONTINUE
      IF (KS(3)) 1210,1210,1180
 1180 IF (TDECP) 1210,1210,1210
CHERE PREVIOUS LINE WAS 1190,1210,1190
C1190 WRITE (6,1200) LL,TDEOP,KAA
 1200 FORMAT (1X27HSYSTEM DOWN AT END OF PHASE,I6,13H FOR DURATION,F10.4
     1,6X7HMISSION,I6)
 1210 CONTINUE
      DNT1=DNT1+TDEOP
      RDT=RDT+TDEOP-DELT
      DELT=TDECP
C
      CALL APPLE
C
CM    WRITE (6,1211)
C1211 FORMAT (2X,'LINE 1211 APPLE')
 1270 CONTINUE
      IF (ICRI) 1280,1290,1280
 1280 REDAD1(JBB)=REDAD1(JBB)+TDEOP
 1290 DNT2=DNT2+DNT1
 1300 IF (DNT2) 1310,1330,1310
 1310 IF(KS(6)) 1330,1330,1330
CHERE PREVIOUS LINE WAS 1325,1330,1325
C1325 WRITE(6,1320) LL,KAA,DNT2
 1320 FORMAT (1X5HPHASE,I5,1X29HTOTAL SYS DOWNTIME IN MISSION,I5,1X3HWAS
     1,F12.4,4H HRS)
 1330 CONTINUE
C
      IF (ICRI) 1350,1350,1340
 1340 IF (ITEMP) 1360,1360,1350
 1350 XCUM=1-ITEMP
      INOABT(JBB)=INOABT(JBB)+1-ITEMP
      INMI(JBB)=INMI(JBB)+1
 1360 CONTINUE
      XNO=INOABT(JBB)
      TNMI=INMI(JBB)
      IF (TNMI) 1380,1380,1370
 1370 RELY=XNC/TNMI
      GO TO 1390
 1380 RELY=0.0
 1390 RELPY=RELPY*RELY
      TT1=ENDPHA-STPHAS
      TT2(JBB)=TT2(JBB)+TT1
      UP1=TT1-DNT1
      UP2(JBB)=UP2(JBB)+UP1
      IF (ISW(N)) 1410,1410,1400
 1400 IAUP2(JBB)=IAUP2(JBB)+1
```

```
 1410  XIAUPP=IAUP2(JBB)
       XAV=XIALPP/XKAA
       IF (KAA-INUM) 1570,1420,1570
CHERE1
C1420 WRITE (6,1430) XAVI
 1430 FORMAT (/47X20HINSTANT AVAILABILITY,5X2X4H IS ,F6.4)
CHERE1
C1440 WRITE (6,1450) LL,JBB,RELY,LL,RELPY
 1450 FORMAT (9X17HRELIABILITY PHASE,I3,1H,,I3,5H, IS ,F6.4,3X25HRELIABI
     1LITY UP TO PHASE   ,I2,4H IS ,F6.4)
C ?
CHERE 1420 IN BELOW LINE SHOULD BE NEXT TO ABOVE WRITE(6,1430)
 1420 RELGA(JBB)=RELPY
      AENDT1=0.0
      AENDT2=0.0
      DO 1520 I=1,KAA
 1460 IF (XTABT(I)-100000.) 1470,1520,1520
 1470 IF (XTABT(I)-TIMA(JBB)) 1480,1520,1520
 1480 AENDT2=AENDT2+TIMA(JBB)-XTABT(I)
      JBB1=JBB-1
      IF (JBB1) 1500,1500,1490
 1490 IF (TIMA(JBB1)-XTABT(I)) 1500,1500,1510
 1500 AENDT1=AENDT1+TIMA(JBB)-XTABT(I)
      GO TO 1520
 1510 AENDT1=AENDT1+TIMA(JBB)-TIMA(JBB1)
 1520 CONTINUE
      TT3=TT3+TT2(JBB)
      UP3=UP3+UP2(JBB)
      REDAD2=REDAD2+REDAD1(JBB)
      RED1=(UP2(JBB)-AENDT1+REDAD1(JBB))/TT2(JBB)
      RED2=(UP3-AENDT2+REDAD2)/TT3
CHERE 1530 LABEL TRANSFERRED BELOW
C1530 WRITE (6,1540) RED1,RED2
 1540 FORMAT (9X16HREADINESS          ,9X4H IS ,F6.4,3X25HREADINESS
     1          ,2X4H IS ,F6.4)
 1530 AVA1=UP2(JBB)/TT2(JBB)
      AVA2=UP3/TT3
CHERE WRITE (6,1550) AVA1,AVA2
 1550 FORMAT (9X23HAVERAGE AVAILABILITY    ,2X4H IS ,F6.4,3X25HAVERAGE AV
     1AILABILITY     ,2X4H IS ,F6.4)
CHERE WRITE (6,1560) XAV
 1560 FORMAT (47X20HINSTANT AVAILABILITY,5X2X4H IS ,F6.4)
 1570 CONTINUE
 1580 KKK=1
      JBB=JBB+1
      T1=SSTIME(LL,N,1)
 1590 RETURN
      END
```

```
C
C
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
C
C
      SUBROUTINE PACK
      COMMON /ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFECP,T3,TIME,T3SUM
      COMMON/BETA/NRO(6,300),IB(6,300,8),NLINE(6)
      COMMON/EXTRA/ KS(20),ISW(31)
      COMMON /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON/NPH/ NSS(6),IFLAG(6),TITLE(6,31),SSTIME(6,31,2),ISS(6,31)
      COMMON/TYP/EX(2,200),ISPARE(3,200),IUSED(3,200),IIUSED(3,200)
      COMMON /MAX/MAXNEQ,MAXTYP,MAXIB,MAXSTD
      COMMON/VDC/VDC(50,6),IUI(200),VMTTR(200,6),TAD2
      COMMON /PACKAP/ IBNUM( 6,500),ISYS( 6),F(200,4)
      COMMON/STAN/ISTB(60,10,6)
      COMMON /CSPARE/ SPR1,SPR2,SPR3,SPR4,SPR5,SPR6,SPR7,SPR8,SPR9
     1,SPR10,SPR11,SPR12,SPR13,SPR14,ITMPOP(200)
      COMMON/XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
C   SULLIVAN ADD
      COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
C   SULLIVAN STOP
      DIMENSICN LOAD(19)
      DIMENSICN DUM(4)
      DIMENSICN IVAL(10)
      DATA IBLANK/4H    /
C
      READ (5,10) KOPT,(KS(I),I=1,13)
      WRITE (6,20) KOPT,(KS(I),I=1,13)
   10 FORMAT (20I4)
   20 FORMAT (1H1,I10,5X 19I4)
C
      READ (5,10) (IFLAG(I),I=1,NPH)
      WRITE (6,30) (IFLAG(I),I=1,NPH)
   30 FORMAT (10I4)
C
      READ(5,40) REPOL,TAD2,XM,XM1
   40 FORMAT(F4.0,F8.0,2F4.0)
   50 FORMAT(20F4.0)
      IF(XM) 35,35,55
   35 XM=1.0
   55 IF(XM1) 36,36,56
   36 XM1=1.
   56 WRITE(6,60) REPOL,TAD2,XM,XM1
   60 FORMAT(1X,4F10.2)
      GO TO (70,90,100,120,130),KOPT
```

```
C
   70 KS(1)=1
      KS(4)=0
   80 KS(3)=0
      KS(2)=0
      KS(2)=1
      KS(5)=0
      KS(6)=0
      KS(7)=0
      KS(8)=0
      KS(9)=0
      KS(10)=0
      GO TO 130
   90 KS(1)=1
      KS(6)=0
      KS(10)=0
      GO TO 110
  100 KS(1)=1
      KS(6)=1
      KS(7)=1
      KS(10)=1
      KS(12)=1
  110 KS(2)=1
      KS(3)=1
      KS(4)=1
      KS(5)=1
      KS(7)=0
      KS(8)=1
      KS(9)=1
      GO TO 130
  120 KS(1)=0
      KS(4)=0
      GO TO 80
C
  130 NEQ=0
      DO 140 I=1,MAXNEQ
      ETIME(I)=100000.
      IEQU(I)=0
  140 CONTINUE
      DO 155 J=1,6
      DO 150 I=1,MAXTYP
      XMTBF(I)=0.0
      VMTTR(I,J)=0.0
  150 XMTTR(I)=0.0
  155 CONTINUE
C
  160 WRITE (6,170)
  170 FORMAT (/11H TYPE   NAME,18X4HMTBF,5X4HMTTR,7X2HDC,8X4HADT1,4X4HADT
```

```
     12)
  180 READ (5,190) I,(DUM(J),J=1,4),X,Y,U,V,W,IDUM
  190 FORMAT (I4,4A4,F8.0,4F4.0,I4)
C   SULLIVAN ADD
      REPTIM(I) = V
C   SULLIVAN STOP
      IF (I) 200,490,200
  200 IF (I-MAXTYP) 220,220,210
  210 WRITE (6,440)
      GO TO 1000
  220 DO 230 J=1,4
  230 F(I,J)=DUM(J)
      IUI(I)=IDUM
      IF (IUI(I)) 240,250,240
  240 READ (5,450) IU,(VDC(IU,ILL),ILL=1,NPH)
  250 IF (Y) 260,280,280
  260 READ (5,50) (VMTTR(I,J),J=1,NPH)
  270 IF (I) 280,490,280
  280 EX(1,I)=V
      EX(2,I)=W
      IF (KS(1)) 310,310,290
  290 WRITE (6,300) I,(F(I,J),J=1,4),X,Y,U,V,W
  300 FORMAT (1XI4,2X4A4,2XF10.1,F10.2,F10.3,2(F8.1))
  310 IF (IUI(I)) 380,380,320
  320 IF (KS(1)) 340,340,330
  330 WRITE (6,460) (VDC(IU,ILL),ILL=1,NPH)
  340 DO 370 ILL=1,NPH
      IF (VDC(IU,ILL)) 360,360,350
  350 VDC(IU,ILL)=(X/VDC(IU,ILL))*XM
      GO TO 370
  360 VDC(IU,ILL)=(X/.0001)*XM
  370 CONTINUE
  380 IF (KS(1)) 410,410,390
  390 IF (Y) 400,410,410
  400 WRITE (6,470) (VMTTR(I,J),J=1,NPH)
  410 IF (XMTBF(I)) 420,430,420
  420 WRITE (6,480) I
      GO TO 1000
  430 IF(U) 435,435,433
  433 XMTBF(I)=XM*(X/U)
  435 XMTTR(I)=Y*XM1
      GO TO 180
  440 FORMAT (9X39HEQUIP TYPES HAVE EXCEEDED MAX ALLOWABLE)
  450 FORMAT (I4,19(F4.0))
  460 FORMAT (14X16HVARY DUTY CYCLE ,4F10.3)
  470 FORMAT (14X16HVARIABLE MTTR   ,4F10.3)
  480 FORMAT (1X4HTYPE,I5,1X13HDEFINED TWICE)
C
```

```
490 WRITE (6,500)
500 FORMAT (/1X15HTYPE   EQUIPMENT)
510 READ (5,10) NTYPE,(LOAD(I),I=1,19)
    IF (LOAD(1)) 520,650,520
520 DO 620 I=1,19
    IF (LOAD(I)) 530,620,530
530 IBM=LOAD(I)
    IF (IBM-500) 560,560,540
540 WRITE (6,550)
550 FORMAT(1X,'EQUIPMENT NUMBER GREATER THAN 500 ******************')
    GO TO 1000
560 IF (IBM-NEQ) 580,580,570
570 NEQ=IBM
580 IF (IEQU(IBM)) 590,610,590
590 WRITE (6,600) IBM
    GO TO 1000
600 FORMAT (1X9HEQUIPMENT,I5,1X34HDEFINED TWICE ********************)
610 CONTINUE
    IEQU(IBM)=NTYPE
620 CONTINUE

    IF (KS(1)) 640,640,630
630 WRITE (6,10) NTYPE,(LOAD(I),I=1,19)
640 NTY=NTYPE
    GO TO 510
C
C OREILLY CHANGE
C 650 WRITE (6,660)
C 660 FORMAT(/1X11HSPARES TYPE,6X4HSHIP,4X6HTENDER,6X4HBASE,12X6HFACTOR)
  650 DO 670 I=1,3
C OREILLY CHANGE - END
    NTYPE=NTY
    DO 670 J=1,NTYPE
670 IUSED(I,J)=0
    READ(5,675)IUNLIM,SX,SPR1,SPR2,SPR3,SPR4,SPR5,SPR6,SPR7,SPR8,SPR9
   1,SPR10,SPR11,SPR12,SPR13,SPR14
675 FORMAT(A4,16X,15F4.0)
    IF(SX-999.) 681,676,681
C
676 CALL SPARES
C
    IF(KS(1)) 740,740,677
677 DO 678 I=1,NTYPE
678 WRITE(6,750) I,(ISPARE(J,I),J=1,3),SX
    GO TO 740
681 IF(SX) 684,682,684
682 SX=1.0
684 IF(IUNLIM-IBLANK)690,720,690
```

```
690  WRITE (6,700)
700  FORMAT (1X41HALL EQUIPMENT TYPES HAVE UNLIMITED SPARES)
     DO 710 I=1,NTYPE
     DO 710 J=1,3
710  ISPARE(J,I)=90000
     GO TO 760
720  DO 740 I=1,NTYPE
     READ (5,10) (ISPARE(J,I),J=1,3)
     BILL=FLOAT(ISPARE(1,I))*SX
     IF(INT(BILL)-BILL) 727,725,727
725  ISPARE(1,I)=BILL
     GO TO 728
727  ISPARE(1,I)=INT(BILL)+1
728  CONTINUE
     IF (KS(1)) 740,740,730
730  WRITE(6,750)I,(ISPARE(J,I),J=1,3),SX
740  CONTINUE
750  FORMAT(5X,I4,2X,3I10,13X,F6.2)
760  WRITE (6,770) NPH
770  FORMAT (1H1,3X28HTHE MISSION WILL BE RUN WITH,I4,7H PHASE ,27HTYPE
    1S IN VARIABLE SEQUENCE.)
C
     DO 777 I=1,6
     DO 776 J=1,10
     DO 775 K=1,60
     ISTB(K,J,I)=0
775  CONTINUE
776  CONTINUE
777  CONTINUE
     DO 990 K=1,NPH
     READ (5,780) XID,LL,NSS(K),ISS(K,NSS(K)+1),SSTIME(K,NSS(K)+1,2)
     ISYS(K)=ISS(K,NSS(K)+1)
780  FORMAT (A4,3I4,F8.0)
     NX=NSS(K)
     N=NX+1
     IF (KS(1)) 820,820,790
790  WRITE (6,810) XID,LL,NSS(K),ISS(K,N),SSTIME(K,N,2)
800  FORMAT (1XA4,3I4,F10.2)
810  FORMAT (/1XA4,3I4,F10.2)
820  TITLE(K,N)=XID
C?
     DO 840 IK=1,NX
     READ (5,780) TITLE(K,IK),KK,MM,ISS(K,IK),SSTIME(K,IK,2)
     IF (KS(1)) 840,840,830
830  WRITE (6,800) TITLE(K,IK),LL,MM,ISS(K,IK),SSTIME(K,IK,2)
840  CONTINUE
C
     DO 850 JA=1,MAXIB
```

```
      DO 850 JB=1,8
      IB(K,JA,JB)=0
      NRO(K,JA)=0
  850 CONTINUE
      IOR=0
      I=0
  860 I=I+1
      READ(5,10) (IVAL(J),J=1,10),IRULE
      IF(IVAL(1).EQ.0) GO TO  990
      IF(IRULE.NE.0)GOTO 930
C
      IF(I.LE.MAXIB) GO TO 880
      WRITE(6,870) MAXIB
  870 FORMAT(1H1,10X,29H# OF GROUP CARDS GREATER THAN,I4)
      STOP
  880 NRO(K,I)=IVAL(1)
C
      DO 890 J=1,8
      IB(K,I,J)=IVAL(J+1)
  890 CONTINUE
      IBNUM(K,IB(K,I,1)-500)=I
      NLINE(K)=I
  900 IF(KS(1)) 860,860,910
  910 WRITE(6,920) NRO(K,I),(IB(K,I,J),J=1,8)
  920 FORMAT(1X,I3,8I4)
      GO TO 860
  930 CONTINUE
      I=I-1
      IOR=IOR+1
C
      IF(IOR.LE.MAXSTD) GO TO 950
      WRITE(6,940) MAXSTD
  940 FORMAT(1H1,10X,36H# OF OPERATE RULE CARDS GREATER THAN,I4)
      STOP
  950 CONTINUE
      DO 960 J=1,10
      ISTB(IOR,J,K)=IVAL(J)
  960 CONTINUE
      IF(KS(1)) 860,860,970
  970 WRITE(6,980) (ISTB(IOR,J,K),J=1,10)
  980 FORMAT(30X,10I4)
      GO TO 860
  990 CONTINUE
 1000 CONTINUE
      RETURN
      END
C
C
```

```
C     ******************************************************************
C
C
      SUBROUTINE EVENT
      COMMON /ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFECP,T3,TIME,T3SUM
      COMMON/TYP/EX(2,200),ISPARE(3,200),IUSED(3,200),IIUSED(3,200)
      COMMON/NPH/NSS(6),IFLAG(6),TITLE(6,31),SSTIME(6,31,2),ISS(6,31)
      COMMON /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON/XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
C  SULLIVAN CHANGE
      COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
C
C
      INTEGER RFKEQ
      NEG = 0
 5    R=ABS(ETIME(1))
      KEQ=1
      DO 15 I=2,NEQ
      RR=ABS(ETIME(I))
      IF (R-RR) 15,15,10
  10  R=RR
      KEQ=I
  15  CONTINUE
      IF (TIME) 160,210,160
 160  IF (IFLAG(LL) - 1) 200,210,210
 200  IF (ETIME(KEQ)) 205,210,210
C
C  FOR A PART THAT HAS GONE TO AN UP STATUS (NEG ETIME) DURING A FLIGHT
C  PHASE, IT CAN'T BE INSTALLED ON THE AIRCRAFT UNTIL THE A/C LANDS ON
C  DECK (BEGINNING OF REPAIR PHASE)
C
 205  IF (ABS(ETIME(KEQ)) - ENDPHA) 207,600,600
 207  ETIME(KEQ) = -ENDPHA - 0.1
      NEG = NEG + 1
C
C  AFTER ADJUSTING THE UPTIME FOR THE PART ABOVE, RESORT THE TIMES TO
C  FIND THE NEXT SCHEDULED EVENT.
C
      IF (NEG - NEQ) 5,220,220
C
C  THE FIRST TIME THROUGH THE SUBROUTINE, THERE ARE NO PARTS IN THE
C  PIPELINE, SO EXIT SUBROUTINE
C
 210  IF (TIME) 600,600,220
C
C  SORT EACH PART TYPE PIPELINE TO FIND EARLIEST TIME
```

```
C
 220   UV = RFITIM(1,1)
       RFKEQ = 1
       DO 300 II= 2,NTYPE
          UV1 = RFITIM(II ,1)
          IF (UV - UV1) 300,300,250
 250      UV = UV1
          RFKEQ = II
 300   CONTINUE
C  COMPARE FIRST PIPELINE TIME (UV) TO FIRST PART ETIME (R)
       IF (UV - R) 340,340,600
C
C  A REPAIR TIME OCCURS BEFORE THE NEXT EVENT
C
 340   IF (UV - ENDPHA) 350,350,600
 350   IF (ISPARE(1,RFKEQ)) 430,430,420
 420   CONTINUE
         IUSED (1,RFKEQ) = IUSED (1,RFKEQ) - 1
 430   CONTINUE
C
C  PIPELINE TIME OCCURS FIRST, PLACE RFI PART IN RETAIL STOCK, DECREASE
C  STOCK USED COUNT BY 1.
C  INCREASE THE PART SUBSCRIPT UP FOR EACH PART STARTING WITH THE 2ND
C
       IJK = NOP(RFKEQ)
       DO 450 KK=1,IJK
         RFITIM(RFKEQ,KK) = RFITIM(RFKEQ,KK +1)
 450    CONTINUE
C
C  SORT THE PIPELINE TO FIND THE NEXT EVENT
       JJ = RFITIM (RFKEQ,1)
C
C  THERE WAS ONLY ONE PART IN THE PIPELINE, NO NEED TO SORT
       IF (IJK - 1) 550,550,520
 520   DO 550 KX = 2,IJK
          JJJ = RFITIM(RFKEQ,KX)
          IF (JJ - JJJ) 550,550,530
 530      RFITIM(RFKEQ,1) = JJJ
          RFITIM (RFKEQ,KX) = JJ
          JJ = JJJ
 550   CONTINUE
C
C  DECREASE THE COUNT OF PARTS IN THE PIPELINE
       NOP(RFKEQ)= NOP(RFKEQ) - 1
C  GO BACK TO SORT THE PIPELINE TIMES AGAIN, FOLLOWED BY A ANOTHER
C  COMPARISON TO SEE IF THE FIRST PIPELINE TIME IS PRIOR TO ETIME
       GO TO 220
 600   CONTINUE
```

```
C
C   THESE NEXT 4 LINES CAN BE ACTIVATED IN ORDER TO EXAMINE REPAIR
C   TIMES OR THE NUMBER OF PARTS IN THE REPAIR PIPLELINE.
C
C      WRITE (6,170) (RFITIM(I,1),I=1,NTYPE)
C170   FORMAT (2X,'RFITIM: ',4F9.1)
C      WRITE (6,185) (NOP(K),K=1,NTYPE)
C185   FORMAT (2X,'NOP ',8I4)
       RETURN
       END
C
C
C    ******************************************************************
C
C
       SUBROUTINE TTE
       COMMON /ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFECP,T3,TIME,T3SUM
       COMMON /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
       COMMON/EXTRA/ KS(20),ISW(31)
       COMMON/NPH/ NSS(6),IFLAG(6),TITLE(6,31),SSTIME(6,31,2),ISS(6,31)
       COMMON /TYP/EX(2,200),ISPARE(3,200),IUSED(3,200),IIUSED(3,200)
       COMMON /DELTA/KKK2
       COMMON /XXX/XXX
       COMMON/VDC/VDC(50,6),IUI(200),VMTTR(200,6),TAD2
       COMMON/GAMMAA/XMTBA,VAR,RELGA(760),TIMA(760),XXT(1442),ITT,ISEED
       COMMON /XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
C SULLIVAN CHANGE
       COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
       REAL UARRAY(2),EARRAY(2)
       NRP  = 0
       CALL LRND(ISEED,UARRAY,2,1,0)
       CALL LEXPN(ISEED,EARRAY,2,1,0)
C
C   K REFERS TO PART NUMBER, J REFERS TO PART TYPE
C
    10 K=KEQ
    20 J=IABS(IEQU(K))
    30 IF (ETIME(K)) 136,31,40
 31    IF (IFLAG(LL) - 1) 36,136,35
C
C   IN THIS CASE, THE MISSION HAS BEGUN WITH THE A/C ON DECK (REPAIR
C   PHASE).  FAILURE TIMES WON'T START UNTIL BEGINNING OF FLIGHT PHASE.
C
  35   ETIME(K) =  ENDPHA + ABS(XXX)*EARRAY(2)
       GO TO 370
C
```

```
C   IN THIS CASE, THE MISSION HAS BEGUN IN THE FLIGHT PHASE.
C   FAILURES CAN BEGIN IMMEDIATELY.
C
 36    ETIME(K) = ABS(XXX)*EARRAY(2) + ETIME(K)
       GO TO 370
C
C   A PART HAS FAILED (ETIME PRIOR TO SUBROUTINE IS POSITIVE)
C
    40 CONTINUE
       IF (ISPARE(1,J)-IUSED(1,J)) 60,60,50
C
C   THERE ARE STILL SOME SPARE PARTS IN INVENTORY, INCREASE JSED COUNT
C
    50 IUSED(1,J)=IUSED(1,J)+1
C
C   NRP IS A MARKER TO SIGNIFY THAT THE FAILED PART WAS REPLACED WITH
C   STOCK ON HAND (NRP = 1 MEANS STOCK USED)
C
       NRP = 1
    60 CONTINUE
C
C   INCREMENT PIPELINE COUNTER (THE NUMBER OF PARTS EITHER IN THE REPAIR
C   PIPELINE OR IN THE ORDER/SHIP PIPELINE)
C
       NOP(J) = NOP(J) + 1
C
C   THIS NEXT LINE WILL ENSURE THAT THE NEXT ETIME WILL BE NEGATIVE,
C   MEANING THAT THE PART IS IN THE FAILURE MODE, AND THE NEXT EVENT
C   WILL OCCUR WHEN THE PART IS RESTORED TO AN UP STATUS.
C
       X = -1.0
C
C   THE FAILED PART WILL EITHER BE REPAIRED OR BCM'ED.  DRAW A RANDOM
C   BCM RATE (BCMDEC), DISTRIBUTED UNIFORM (0,(2*BCM(J))
C
       BCMDEC = UARRAY(1) *2* BCM(J)
C
C   IF A UNIFORM(0,1) RANDOM NUMBER IS LESS THAN THE BCMRATE, BCM PART.
C
       IF (UARRAY(2) - BCMDEC) 115,115,118
C
C   PART WILL BE BCM'ED, COMPUTE EXPONENTIAL ORDER/SHIP TIME
C
 115   ADT = EARRAY(1) * SRTIM(J)
       GO TO 125
C   PART WILL BE REPAIRED, COMPUTE EXPONENTIAL REPAIR TIME
C
 118   ADT = EARRAY(1) * REPTIM(J)
```

```
C
C   PIPELINE TIMES ARE ASSIGNED TO PARTS PLACED IN THE REPAIR OR
C   ORDER/SHIP PIPELINE, AS SOON AS PHASE ENDS (A/C LANDS).
C
 125  RFITIM(J,NOP(J)) = ADT + ENDPHA
C
C   SORT PIPELINE TIMES, PLACE EARLIEST TIME FIRST, FOR EACH TYPE
C
      JJ = RFITIM(J,1)
      IJ = NCP(J)
      IF (IJ.EQ.1) GO TO 130
      DO 130 KX = 2,IJ
        JJJ = RFITIM(J,KX)
        IF (JJ - JJJ) 130,130,128
 128    RFITIM(J,1) = JJJ
        RFITIM (J,KX) = JJ
        JJ = JJJ
 130  CONTINUE
C
C   IF THE FAILED PART WAS REPLACE WITH ONE FROM INVENTORY, THERE WILL
C   NO DOWNTIME WHILE THE PIPELINE  PRODUCES A SPARE PART FOR THE NEXT
C   EXPECTED UP TIME.  WAITING INVOLVES ONLY PART REPLACEMENT TIME.
C
      IF (NRP) 140,140,137
C
C   FOR A PART ENTERING SUBROUTINE WITH NEGATIVE ETIME (ARRIVING IN
C   DOWN STATUS), NEXT LINE ENSURES POSITIVE ETIME NEXT EVENT
C
 136  X = 1.0
      ETIME(K) = X*(ABS(XXX)*EARRAY(2) + ENDPHA)
      GO TO 370
C
C   NO DELAY (ADT) IS INCURRED FOR REPLACING A FAILED PART THAT HAS A
C   SPARE AVAILABLE.  NEXT EVENT TIME IS COMPUTED.
C
C
  137 ETIME(K) = X*(ABS(XXX)*EARRAY(2) + ENDPHA)
      GO TO 370
C
C   NO SPARES WERE AVAILABLE, UPTIME MUST INCLUDE A DELAY TIME.
C
 140  ETIME(K)=X*(ABS(XXX)*EARRAY(2)+ENDPHA+ADT)
  370 CONTINUE
C     WRITE (6,499) XXX
C 499 FORMAT (2X,'XXX ',F10.4)
      RETURN
      END
C
```

```
C
C     ******************************************************************
C
C
      SUBROUTINE STNDBY
      COMMON /ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFEOP,T3,TIME,T3SUM
      COMMON /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON /XXX/XXX
      COMMON/STAN/ISTB(60,10,6)
      COMMON /XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NUAC
      DO 170 I=1,50
      IF (ISTB(I,1,LL)) 10,180,10
C
   10 INDEX=1
      DO 50 J=2,10
      KK=ISTB(I,J,LL)
      IF (KK) 30,60,20
   20 IF (ETIME(KK)) 40,50,50
C
   30 KK=IABS(KK)
      IF (ETIME(KK)) 40,40,50
   40 INDEX=0
      GO TO 60
   50 CONTINUE
C
   60 K=IABS(ISTB(I,1,LL))
C
      ISO=ISTB(I,1,LL)
C
   70 IF (ETIME(K)) 170,170,80
   80 IF (ETIME(K)-100000.) 120,90,120
   90 IF (INDEX) 170,110,100
  100 IF (ISO) 170,170,150
  110 IF (ISO) 150,170,170
  120 IF (INDEX) 170,140,130
  130 IF (ISO) 160,170,170
  140 IF (ISO) 170,170,160
C
  150 IABC=IABS(IEQU(K))
      XXX=XMTBF(IABC)
      KEQ=K
      CALL TTE
      GO TO 170
C
  160 ETIME(K)=100000.
  170 CONTINUE
```

```
  180 RETURN
      END
C
C
C  ****************************************************************
C
C
      SUBROUTINE STATUS
      COMMON /ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFEOP,T3,TIME,T3SUM
      COMMON/BETA/NRO(6,300),IB(6,300,8),NLINE(6)
      COMMON/EXTRA/ KS(20),ISW(31)
      COMMON /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON/NPH/ NSS(6),IFLAG(6),TITLE(6,31),SSTIME(6,31,2),ISS(6,31)
      COMMON /XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
      COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
C
      KID=0
      NL1=NLINE(LL)
   10 DO 130 K=1,NL1
   12 KT=IB(LL,K,1)
   14 IF(KID-KT)16,18,16
   16 ISUM=0
C
   18 IF(NRO(LL,K))130,130,20
   20 DO 60 J=2,8
   30 KK=IABS(IB(LL,K,J))
      IF (KK) 70,70,40
   40 IF (ETIME(KK)) 60,60,50
   50 ISUM=ISUM+1
   60 CONTINUE
   70 CONTINUE
      IF (ISUM-NRO(LL,K)) 80,90,90
   80 ETIME(KT)=-1.
      GO TO 100
   90 ETIME(KT)=1.
C  SULLIVAN CHANGE
C 100 IF(KS(12))125,125,110
  100 IF (KS(12))125,125,125
C 110 WRITE(6,120)KT,ETIME(KT)
C 120 FORMAT(1X3HKK=,I5,7H ETIME=,F10.5)
C  SULLIVAN STOP
  125 KID=KT
  130 CONTINUE
      N=NSS(LL)+1
      DO 160 I=1,N
      J=ISS(LL,I)
```

```
      IF  (ETIME(J))  140,140,150
C SULLIVAN CHANGE ISW(I)  TO REAL FORMAT
  140 ISW(I)=-1.0
      GO  TO  160
  150 ISW(I)=1.0
C  SULLIVAN  STOP
  160 CONTINUE
      KZZ=0
      RETURN
      END
C
C
C   *************************************************************************
C
C
      SUBROUTINE  APPLE
      DIMENSION  IPRNT(50),ICHLD(50),MKBA(100)
      COMMON  /ALPHA/ DNT2, ENDPHA, ICRI, IFF, IFR, INUM, IOPT, JBB, KEQ, KKK, KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFEOP,T3,TIME,T3SUM
      COMMON/BETA/NRO(6,300),IB(6,300,8),NLINE(6)
      COMMON  /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON  /TIGAP/  UP4,XNUM,BAPRIN,AVA,XPCAP,RUNID(19),TYCOON(500)
     +,COUNTB(500),XTCUM
      COMMON/RUNAP/ITEMP2,DELT,ISSA(31),ISSC
      COMMON/NPH/NSS(6),IFLAG(6),TITLE(6,31),SSTIME(6,31,2),ISS(6,31)
      COMMON  /PACKAP/  IBNUM( 6,500),ISYS( 6),F(200,4)
      COMMON/XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
C
      IF(BAPRIN)790,90,90
   90 JCOUNT=0
C
  100 IPTR=0
      L=LL
      IF(ITEMP2)240,105,107
  105 K=IBNUM(L,ISYS(L)-500)
      GOTO 108
C ??????????????
  107 KSS=ISSA(ISSC)
      K=IBNUM(L,ISS(L,KSS)-500)
  108 KID1=IB(L,K,1)
  110 NN=2
C
  120 DO  210  N=NN,8
      IGRP=IB(L,K,N)
      IF(IGRP)240,212,140
  140 IF  (ETIME(IGRP))  150,150,210
  150 IF  (IGRP-500)  170,170,160
```

```
C
  170 IF (JCCUNT) 240,200,180
  180 DO 190 I=1,JCOUNT
      IF (MKBA(I)-IGRP) 190,210,190
  190 CONTINUE
  200 CONTINUE
      JCOUNT=JCOUNT+1
      MKBA(JCCUNT)=IGRP
  210 CONTINLE
  212 IF(K-1)220,220,214
  214 KID2=IB(L,K-1,1)
      IF(KID1-KID2)220,216,220
  216 K=K-1
      GOTO 108
C
  220 IF (IPTR) 240,260,230
C
  230 K=IPRNT(IPTR)
      KID1=IB(L,K,1)
      NN=ICHLD(IPTR)
      IPTR=IPTR-1
      GOTO 120
C
C
  160 IF (N-8) 165,167,240
C
  165 IPTR=IPTR+1
      IPRNT(IPTR)=K
      ICHLD(IPTR)=N+1
  167 K=IBNUM(L,IGRP-500)
      GOTO 108
C
  240 WRITE (6,250)
  250 FORMAT (12H APPLE ERROR)
      GO TO 300
C
  260 IF(ITEMP2) 240,265,262
  262 ISSC=ISSC-1
      IF (ISSC) 240,265,100
  265 FCCUNT=FLOAT(JCOUNT)
      IF (ITEMP2) 270,270,280
C
  270 DO 275 I=1,JCOUNT
  275 TYCOON(MKBA(I))=TYCOON(MKBA(I))+DELT/FCCUNT
      GOTO 300
C
C
  280 DO 290 I=1,JCOUNT
```

```
  290 COUNTB(MKBA(I))=COUNTB(MKBA(I))+1/FCOUNT
  300 CONTINUE
      RETURN
C
  790 CONTINUE
      WRITE (6,800) (RUNID(I),I=1,19)
  800 FORMAT(1H1,3X,19A4//)
      WRITE (6,810)
  810 FORMAT(32X,19HCRITICAL EQUIPMENTS//32X,18HUNAVAILABILITY AND/    27
     1X25HPERCENT OF UNAVAILABILITY//)
      WRITE (6,820)
  820 FORMAT (24X4HNAME,17X7HNUM HRS,11X5HUNAVA,2X7HPERCENT,6X8HEQU TYPE
     1,5X7HEQU NUM/)
C
      IF (AVA-1.) 830,880,830
  830 TR=TYCCON(1)
      INDEX=1
      DO 850 I=2,NEQ
      TRR=TYCCON(I)
      IF (TR-TRR) 840,850,850
  840 TR=TRR
      INDEX=I
  850 CONTINUE
      TYCUM=TYCOON(INDEX)/TT3
      TYCUM2=TYCOON(INDEX)/(TT3-UP4)*100.
      IF (TYCCON(INDEX)) 860,880,860
  860 IXX=IABS(IEQU(INDEX))
      WRITE (6,870) (F(IXX,J),J=1,4),TYCOON(INDEX),TYCUM,TYCUM2,IXX
     1,INDEX
  870 FORMAT (20X4A4,F20.4,4XF8.4,F8.2,8XI4,10XI4)
C
      TYCOON(INDEX)=0.0
      GO TO 830
  880 WRITE (6,800) (RUNID(I),I=1,19)
      WRITE (6,910)
  910 FORMAT(32X,19HCRITICAL EQUIPMENTS//32X,17HUNRELIABILITY AND/    27X
     127HPERCENT OF MISSION FAILURES//)
      WRITE (6,920)
  920 FORMAT (12X11HDESCRIPTION,8X3HNO.,6X6HUNREL ,3X7HPERCENT,2X13HEQUI
     1P  EQUIP /28X8HFAILURES,22X10HTYPE   NO.)
      IF (XPCAP-1.) 930,1090,930
C
  930 INEWA=0
      DO 950 I=1,NEQ
      IF (COUNTB(I)) 950,950,940
  940 INEWA=INEWA+1
      MKBA(INEWA)=I
  950 CONTINUE
```

```
C
      TOTAL=XNUM-XTCUM
  955 IF (INEWA-1) 1010,975,952
  952 INDEX=MKBA(1)
      NN=1
      TR=COUNTB(INDEX)
      DO 970 I=2,INEWA
      IF (TR-COUNTB(MKBA(I))) 960,970,970
  960 INDEX=MKBA(I)
      NN=I
      TR=COUNTB(INDEX)
  970 CONTINUE
  977 UNREL=TR/XNUM
      PERC=TR/TOTAL*100.
      IND=IABS(IEQU(INDEX))
      WRITE(6,990) (F(IND,J),J=1,4),TR,UNREL,PERC,IND,INDEX
  990 FORMAT (9X4A4,3XF6.1,5XF6.4,3XF6.2,4XI4,3XI4)
      MKBA(NN)=MKBA(INEWA)
      INEWA=INEWA-1
      GOTO 955
C
  975 INDEX=MKBA(1)
      TR=COUNTB(INDEX)
      GOTO 977
C
 1010 JNUM=IFIX(XNUM)
      WRITE (6,1020) JNUM
 1020 FORMAT (//9X19HTOTAL NO. MISSIONS=,I4)
      ITOTAL=TOTAL
      WRITE (6,1030) ITOTAL
 1030 FORMAT (9X27HTOTAL NO. MISSION FAILURES=,I4)
 1090 RETURN
      END
C
C
C  ****************************************************************
C
C
      SUBROUTINE SPARES
C
      COMMON /ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,IFFECP,T3,TIME,T3SUM
      COMMON /N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON /TYP/EX(2,200),ISPARE(3,200),IUSED(3,200),IIUSED(3,200)
      COMMON /CSPARE/ SPR1,SPR2,SPR3,SPR4,SPR5,SPR6,SPR7,SPR8,SPR9
     1,SPR10,SPR11,SPR12,SPR13,SPR14,ITMPOP(200)
C    CREILLY ACD
```

```
      COMMON/XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
      COMMON /KSPARE/JTIME,TOTSPR
C    SULLIVAN CHANGE
      COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
      WRITE (6,999) XFLAG
 999  FORMAT (2X,'XFLAG: ',F4.1)
C
C   SULLIVAN CHANGE
C
C   THIS NEXT LINE WILL READ IN ACIM INVENTORY LEVELS (OR ANY OTHER
C   ARBITRARY LEVELS CHOSEN)
C
      READ (5,6) (ISPARE(1,J),J=1,NTYPE)
      IF(XFLAG-1.) 5,2,3
    2 CALL ASPARE
C   2 CALL MSPARE
C
      WRITE (6,9)
      GO TO 101
C   3 CALL GSPARE
  3   CALL RIMAIR
      WRITE (6,11)
      GO TO 101
C
C     WRITE (6,22)
C     GO TO 101
C   5 CUT=SPR1
    6 FORMAT (35I2)
  5   CONTINUE
      GO TO 101
C   SULLIVAN STOP
C     XAVAIL=.9
C     XBUDL=.85*BUDGET
C     WRITE(6,300)XBUDL
  300 FORMAT(/1X,5HBUDL ,F8.3)
      XBUDH=1.05*BUDGET
      WRITE(6,301)XBUDH
  301 FORMAT(/1X,5HBUDH ,F8.2)
      HIGH=1.
      LOW =0.0
C
C    SULLIVAN CHANGE
    9 FORMAT (/1X,31HSPARES BEING COMPUTED USING ASO)
      WRITE(6,303)XAVAIL
   11 FORMAT (/1X,34HSPARES BEING COMPUTED USING RIMAIR)
C    SULLIVAN STOP
C    OREILLY STOP
      DO 10 I=1,NTYPE
```

```
      ITMPOP(I)=0
   10 CONTINUE
      DO 20 I=1,NEQ
      ITMPOP(IEQU(I))=ITMPOP(IEQU(I))+1
   20 CONTINUE
   25 DO 90 I=1,NTYPE
      EX90DD=((8766./XMTBF(I))/4.)*ITMPOP(I)
C     WRITE (6,29) I,EX90DD
   29 FORMAT(/1X,16HEX90DD FOR ITEM ,I4,4H IS ,F8.0)
      IF(EX90DD-1.) 60,30,30
C
   30 PRBSUM=EXP(-EX90DD)
      DUM=PRBSUM
      KFACT=1
      K=0
   40 K=K+1
      KFACT=KFACT*K
      PRBSUM=PRBSUM+DUM*(EX90DD**K)/KFACT
C     WRITE(6,303)XAVAIL
  303 FORMAT(/1X,9HXAVAIL = ,F6.4)
      IF(PRBSUM-XAVAIL) 40,50,50
   50 ISPARE(1,I)=K
      GO TO 90
   60 IF(4.*EX90DD-CUT) 80,80,70
C
   70 ISPARE(1,I)=1
      GO TO 90
   80 ISPARE(1,I)=0
   90 CONTINUE
      XSUM=0.0
      DO 95 I=1,NTYPE
      XSUM1=ISPARE(1,I)*COST(I)
      XSUM=XSUM+XSUM1
   95 CONTINUE
      WRITE(6,302)XSUM
  302 FORMAT(/1X,5HXSUM ,F8.2)
C     IF (XSUM-XBUDL) 200,97,96
C  96 IF (XSUM-XBUDH) 97,97,210
      IF (XSUM-XBUDH) 200,200,205
   97 WRITE(6,98) XAVAIL
   98 FORMAT(/1X,44H,FSLIP ALLOWS CONSTRAINED BY BUDGET, XAVAIL= ,F8.6)
   99 DO 100 I=1,NTYPE
      DO 100 J=2,3
      ISPARE(J,I)=0
  100 CONTINUE
C    OREILLY ADD
      WRITE (6,22)
   22 FORMAT(/1X11HSPARES TYPE,6X4HSHIP,4X6HTENDER,6X4HBASE,12X6HFACTOR)
```

```
  101 RETURN
  200 IF(XAVAIL-.9) 97,99,99
C 205 XLOW=XAVAIL
C     XAVAIL=(HIGH+XAVAIL)/2.
  205 IF(XAVAIL-0.1) 208,208,206
  206 XAVAIL=XAVAIL-.05
      WRITE(6,303)XAVAIL
      GO TO 25
  208 DO 215 I=1,NTYPE
      J=1
      ISPARE(J,I)=0
  215 CONTINUE
      GO TO 97
C 210 HIGH=XAVAIL
C     XAVAIL=(XLOW+XAVAIL)/2.
C     GO TO 25
C   CREILLY STOP
      END
C
C
C  ***************************************************************************
C
C   SULLIVAN ADDS
C
      SUBROUTINE ASPARE
C THIS PROGRAM CALCULATES AVCAL SPARES USING ASO POLICY
C
      COMMON/ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPT,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED2
     2,REPLY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,CFFEOP,T3,TIME,T3SUM
      COMMON/N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON/XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
      COMMON/KSPARE/JTIME,TOTSPR,COMB(9999),COMBA(9999),SER(100)
      COMMON/TYP/EX(2,200),ISPARE(3,200),IUSED(3,200),IIUSED(3,200)
      COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
      REAL TCCST,MRP,V(200)
      INTEGER AVCAL,RP,AA
C
C  INITIALIZE VARIABLES
      TCOST=0.0
      WRITE(6,1)
 1    FORMAT(/1X,46HSPARES WILL BE DETERMINED USING ASOINST POLICY)
      DO 5 I=1,NTYPE
      DO 5 J=1,3
        ISPARE(J,I)=0
 5    CONTINUE
      DO 91 K=1,NTYPE
      AA=0
```

```
C
      AVCAL = 0
      RP = 0
      I= 0
      QTRDEM = (FLOAT((JTIME)*NPET(K)*NOAC))/XMTBF(K)
      WRITE (6,991)QTRDEM,JTIME,NPET(K),NOAC,XMTBF(K)
 991  FORMAT(2X,'QTRDEM ',F9.4,2X,I5,1X,I2,2X,I2,2X,F9.4)
      BCMR = QTRDEM*BCM(K)
      MRP =( QTRDEM*(1.0-BCM(K)))
      MRP = MRP*REPTIM(K)/2160.
      IF (MRP.LT.0.11) GO TO 40
         IF (MRP.LT.20.0 ) GO TO 10
            RP = 1.2817 * (MRP**0.5) + MRP + 0.5
            GC TO 40
 10    E = EXP (-MRP)
      X=E
      Y=E
 15    IF (X.GE.0.895) GO TO 20
      I=I+1
      Y=Y*MRP/FLOAT(I)
      X=X+Y
      GO TO 15
 20   RP=I
 40   CONTINUE
      IF (RP.EQ.0) GO TO 70
         IF (BCMR.LT.1.0) GO TO 50
         IF (IFIX (BCMR + 0.5) .LE.1) AA = 1
       IF (IFIX(BCMR+0.5).GT.1) AA=IFIX(BCMR+0.5)
            GO TO 60
 50    CONTINUE
      AA=0
 60   CONTINUE
      GO TO 80
 70   CONTINLE
      IF (COST(K).LT.5000.) GO TO 75
       IF (BCMR.LT.0.5) GO TO 80
         IF (IFIX ( BCMR + 0.5) .LE. 1) AA =1
         IF (IFIX(BCMR+0.5).GT.1) AA=IFIX(BCMR+0.5)
         GO TC 80
 75    CONTINUE
      IF (BCMR .LT. 0.34) GO TO 80
         IF (IFIX (BCMR + 0.5) .LE. 1) AA = 1
         IF (IFIX(BCMR+0.5).GT.1)AA=IFIX(BCMR+0.5)
 80   CONTINUE
      AVCAL = RP + AA
      TCOST=TCOST+COST(K)*FLOAT(AVCAL)
      ISPARE(1,K)=AVCAL
 91   CONTINUE
```

```
C
C     WRITE(6,101)
C101  FORMAT(/1X,29HALL SPARES HAVE BEEN COMPUTED)
      WRITE (6,191) TCOST
 191  FORMAT (2X,'TOTAL COST ASOMAN: ',F12.1)
      RETURN
      END
C
C
C   ***************************************************************
C
C
      SUBROUTINE RIMAIR
C
C    THIS PROGRAM UTILIZES RIMAIR OPTIMIZATION TO COMPUTE AVCAL
C
      COMMON/ALPHA/DNT2,ENDPHA,ICRI,IFF,IFR,INUM,IOPT,JBB,KEQ,KKK,KZZZ
     1,KK1,KS1,LL,LLLAST,NEQ,NPH,NTYPE,NUM,REDAD2,REDAD1(760),RELP,RED22
     2,RELPY,REPOL,STPHAS,TP,T1,XCUM,TT3,UP3,OFFEOP,T3,TIME,T3SUM
      COMMON/N/IEQU(500),KEQU(500),ETIME(1000),XMTBF(200),XMTTR(200)
      COMMON/XSPARE/XFLAG,BUDGET,COST(201),RFITIM(31,20),NPET(31),NOAC
      COMMON/KSPARE/JTIME,TOTSPR,COMB(9999),COMBA(9999),SER(100)
      COMMON/TYP/EX(2,200),ISPARE(3,200),IUSED(3,200),IIUSED(3,200)
      COMMON/YSPARE/BCM(200),SRTIM(200),EL,ESS,RET,REPTIM(200),NOP(200)
      INTEGER OPTMAL
      REAL MRP,V(200),EL
C
C
C    INITIALIZE VARIABLES
      TCOST= 0.
      DO 3 I=1,NTYPE
        DO 3 J=1,3
          ISPARE(J,I)=0
  3   CONTINUE
C     ESS=1.0
 5    DO 950 KK=1,NTYPE
      OSTP = SRTIM(KK)/24.0
      OSTW = SRTIM(KK)/24.0
      BCMRAT = BCM(KK)
      BCMWAR = BCM(KK)
      QTRDEM = (FLOAT((JTIME)*NPET(KK)*NOAC))/XMTBF(KK)
      BCMP=QTRDEM*BCMRAT
      BCMW = QTRDEM * BCMWAR
      MRP = REPTIM(K) * QTRDEM * (1. - BCMRAT)/2160.
      COSTK=CCST(KK)
      WP = MRP + (OSTW + RET )*BCMWAR *QTRDEM/90.
      IF (WP. EQ.0.0) GO TO 5
C    BEGIN EVALUATION
```

```
      OLP=((558.61*BCMP)/COSTK)**0.5
      OLW=((558.61*BCMW)/COSTK)**0.5
      END = BCMP * (1. - (RET/90.)-(OSTW/90.)) + (BCMP * OSTP/90.)
C     END MUST BE ZERO OR GREATER
      IF (END.LT.0.0)  END =0.0
C     CALCULATE BASIC PIPELINE
      GR = WP + END
C     CALL APPROPRIATE SUBROUTINE BASED ON WARTIME PIPELINE
      IF (WP.GE.5.) GO TO 10
      CALL PCISON (WP,GR,EL,COSTK,QTRDEM,ESS,OPTMAL,OLP)
      GO TO 20
 10   CALL NORMAP (OLP,EL,WP,COSTK,QTRDEM,ESS,GR,CPTMAL)
 20   CALL AVCAL (GR,OLP,OPTMAL,COSTK,CCOST)
      TCOST = TCOST + CCOST
C     IF (OPTMAL.EQ.0.) GO TO 5
      ISPARE(1,KK) = OPTMAL
 950  CONTINUE
      WRITE (6,335) TCOST
 335  FORMAT(2X,'TOTAL COST RIMAIR: ',F13.0)
      RETURN
      END
C
C
C
      SUBROUTINE POISON (WP,GR,EL,COSTK,QTRDEM,ESS,OPTMAL,OLP)
      INTEGER S,OPTMAL,MAX
      REAL EL,MN,ESS,WP,GR,QTRDEM,A,PS3,CT,PS,OLP
      MN = 0.0
      A = 0.99
      MAX = 0
      PS3 = EXP(-GR)
      CT=PS3
 5    IF (CT.GE.A) GO TO 7
        MAX = MAX + 1
        PS3 = PS3*GR/FLOAT(MAX)
        CT = CT + PS3
        GO TO 5
 7    CONTINUE
      MAX = MAX + OLP - 1 + 0.5
C     TRUNCATE WARTIME PIPELINE TO INTEGER VALUE
      S=WP
C     CALCULATE POISSON DISTRIBUTION OF MEAN WARTIME PIPELINE
      PS = EXP (-WP)
C     CALCULATE POISSON DISTRIBUTION OF BASIC PIPELINE
      PS2 = EXP (-GR)
C     ASSIGN CS POISSON OF GR
      CS = PS2
C     CALCULATE MINIMUM STOCK
```

```
      IF (QTRDEM.NE.0.0) MN=(EL*COSTK)/(QTRDEM*ESS)
C    OPTIMIZATION LOOP
      IF (S.EQ.0) GO TO 15
      DO 10  I=1,S
         PS =  (PS*WP)/FLOAT(I)
         PS2=(PS2*GR)/FLOAT(I)
 10      CS=CS+PS2
C    COMPARE WARTIME PIPELINE POISSON VALUE WITH MINIMUM STOCK,
C    IF LOWER, BRANCH
 15   IF (PS.LT.MN) GO TO 40
 20      IF (CS.GE.A)   GO TO 30
            IF (PS.LT.MN) GO TO 40
               S = S + 1
C   CALCUALTE NEW POISSON VALUES AND CS
               PS=(PS*WP)/FLOAT(S)
               PS2=(PS2*GR)/FLOAT(S)
               CS=CS+PS2
               GO TO 20
 30         IF (PS.LT.MN) GO TO 40
               IF (S.GE.MAX) GO TO 40
               S = S +1
               PS  = PS*WP/FLOAT(S)
               GO TO 30
 40    CONTINUE
       OPTMAL  = S
       RETURN
       END
C
C    NORMAL APPROXIMATION TO POISSON
C
      SUBROUTINE NORMAP (OLP,EL,WP,COSTK,QTRDEM,ESS,GR,OPTMAL)
      INTEGER OPTMAL,LMN,NNP
      REAL ESS,PI,OLP,OLW,WP,COST(200),QTRDEM,GR
      PI=3.14159
C    CALCULATE APPROX OF MIN STOCK USING NORMAL APP
      AMN=(EL*COSTK*(2*PI*WP)**0.5)/(QTRDEM*ESS)
C   COMPARE APPROX TO ONE, BRANCH IF LESS OR EQUAL
      IF (AMN.LE.1) GO TO 10
C   SET OPTMAL EQUAL TO ZERO, RETURN TO MAIN
         OPTMAL = 0
         RETURN
C   CALCULATE LMN AND NNP (ROUNDUP)
 10     LMN=WP + 1 + ((-2)*WP*ALOG(AMN))**0.5
        NNP=2.33*(GR**0.5) + GR + OLP
        IF (NNP.LT.LMN) GO TO 20
C   SET OPTMAL EQUAL TO LMN, RETURN TO MAIN
      OPTMAL =LMN
      RETURN
```

```
C     SET OPTMAL EQUAL TO NNP, RETURN TO MAIN
 20     OPTMAL = NNP
        RETURN
        END
C
C      COMPUTE AVCAL
C
        SUBROUTINE AVCAL (GR,OLP,OPTMAL,COSTK,CCOST)
        INTEGER OPTMAL,MIN,IAVCAL
        REAL A,GR,OLP,OLW,COST(200),CCOST
        A=0.5
C     COMPARE GROSS REMOVALS TO CONSTANT, IF LOW, BRANCH
        IF (GR.LT.A) GO TO 10
C     COMPARE OPERATING LEVEL TO ONE, SET EQUAL TO ONE IF LOW
           IF (OLP.LT.1.) OLP=1.
 10     MIN=OLP+GR+0.5
C     COMPARE OPTMAL TO MIN, BRANCH IF EQUAL OR GREATER
        IF(OPTMAL.GE.MIN) GO TO 20
C     SET IAVCAL EQUAL TO MIN
           IAVCAL =MIN
           GO TO 30
C     SET IAVCAL EQUAL TO OPTMAL
 20     IAVCAL=OPTMAL
C     CALCULATE COST AND RETURN TO MAIN
 30     CCOST=FLOAT(IAVCAL)*COSTK
        OPTMAL=IAVCAL
        RETURN
                 END
```

LIST OF REFERENCES

1. Chief of Naval Operations Instruction 5442.4H (OPNAVINST 5442.4H), "Aircraft and Training Devices Material Condition Definitions, Mission-Essential Subsystems Matrices, and Mission Descriptions", pp. 101-104, 20 May 1983.

2. Leather, J.E., An Evaluation of the Effect of Spares Allowance Policy Upon Ship Availability and Reliability, M.S. Thesis, Naval Postgraduate School, Monterey, CA., September 1980.

3. O'Reilly, P.J., An Evaluation of Allowance Determination Using Operational Availability, M.S. Thesis, Naval Postgraduate School, Monterey, CA, June, 1982.

4. Naval Ship Engineering Center, Report No. 6112B-036-78, TIGER Manual, February, 1979.

5. Naval Postgraduate School Report NPS55-81-005, Naval Postgraduate School Random Number Package LLRANDOMII, Lewis, P.A.W., Uribe, L., February 1981.

6. Boatwright, B.O., RIMAIR VS. Current ASO Policy: A Comparative Analysis of Two Methods For Determining Avcal Stockage Levels, pp. 76-84, M.S. Thesis, Naval Postgraduate School, Monterey, CA, September, 1983.

7. Naval Operations Analysis, 2nd ed., pp. 262-272, United States Naval Institute, 1979.

8. Fleet Material Support Office, ALRAND Working Memorandum 352, Aviation Supply Support of Operating Forces, 14 March 1980.

9. Ross, S.M., Introduction to Probability Models, New York, Academic Press, 1980.

10. Naval Aviation Supply Office Instruction 4423.32 CH-1, ASO Provisioning Manual, pp. III-3-1, III-3-15, 27 November 1978.

11. Mitchell, M.L., A Retail Level Inventory Model for Naval Aviation Repairable Items, M.S. Thesis, Naval Postgraduate School Thesis, Monterey, CA, March, 1983.

12. Aviation Supply Office letter to Naval Supply Systems Headquarters ACA-1: JPB: mec 4790, Subj: Turn Around Time Constraints, 2 November 1977.

13. Chief of Naval Operations Instruction 441.12A, Supply Support of the Operating Forces, Enclosure (5), pg. 6, 9 August 1973.

14. Fleet Material Support Office Report 155, RIM-AIR Study, pg. 1, Operations Analysis Department, 30 June 1983.

15. Department of Defense Instruction 4140.46, Standard Stockage Policy for Repairable Secondary Items at the Intermediate and Consumer Levels of Inventory, pp. 3-4, 7 April 1978.

16. Naval Seas Systems Command, Availability Centered Inventory Model Consumer Level Allowance Development Handbook, May 1983.

17. McDonnell, J., LAMPS MK III Pack-Up Kit Spares Selection As Depicted By the Availability Centered Inventory Model (ACIM), M.S. Thesis, Naval Postgraduate School, Monterey, CA, March, 1984.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Assoc. Professor F.R. Richards, Code 55Rh Naval Postgraduate School Monterey, California 93943	2
4.	Assoc. Professor A.W. McMasters, Code 55Mg Naval Postgraduate School Monterey, California 93943	1
5.	Mr. Peter Evanovich Center for Naval Analysis 2000 North Beauregard St. Alexandria, Virginia 22311	1
6.	LCDR Mark David Sullivan 4921 Whitewood Lane Virginia Beach, Virginia 23464	2